Detecting and Preventing IP-spoofed Distributed DoS Attacks

Yao Chen[1], Shantanu Das[1], Pulak Dhar[2], Abdulmotaleb El Saddik[1], and Amiya Nayak[1]
(Corresponding author: Shantanu Das)

School of Information Technology and Engineering, University of Ottawa[1]
800 King Edward Avenue, Ottawa, ON K1N 6N5, Canada (Email: shantdas@site.uottawa.ca)
Cistel Technology Inc., 30 Concourse Gate, Unit 40, Ottawa, ON K2E 7V7, Canada[2]

(Received Aug. 9, 2006; revised and accepted Nov. 8, 2006)

Abstract

In this paper, we explore mechanisms for defending against Distributed Denial of Service (DDoS) attacks, have become one of the major threats to the operation of the Internet today. We propose a novel scheme for detecting and preventing the most harmful and difficult to detect DDoS Attacks—those that use IP address spoofing to disguise the attack flow. Our scheme is based on a firewall that can distinguish the attack packets (containing spoofed source addresses) from the packets sent by legitimate users, and thus filters out most of the attack packets before they reach the victim. Unlike the other *packet-marking* based solutions, our scheme has a very low deployment cost; We estimate that an implementation of this scheme would require the cooperation of only about 20% of the Internet routers in the marking process. The scheme allows the firewall system to configure itself based on the normal traffic of a Web server, so that the occurrence of an attack can be quickly and precisely detected. We have extensively tested our scheme by simulating DDoS attacks with up to several thousand attackers and the experimental results show that more than 90% of attack packets can be effectively filtered-out without much affecting the flow of legitimate packets to the victim Web-server.

Keywords: Distributed denial-of-service attacks, firewall, IP address spoofing, packet filtering

1 Introduction

Today, the Internet is an essential part of our everyday life and many important and crucial services like banking, shopping, transport, health, and communication are partly or completely dependent on the Internet. According to recent sources [12, 13] the number of hosts connected to the internet has increased to almost 400 million and there are currently more than 1 billion users of the Internet. Thus, any disruption in the operation of the Internet can be very inconvenient for most of us.

As the Internet was originally designed for openness and scalability without much concern for security, malicious users can exploit the design weaknesses of the internet to wreak havoc in its operation. Incidents of disruptive activities like e-mail viruses, computer worms and denial-of service attacks have been on the rise ([6] reports an increase of such incidents from 252 in 1990 to 137,529 in 2003). The incidents which has raised the most concern in recent years are the denial-of-service(DoS) attacks whose sole purpose is to reduce or eliminate the availability of a service provided over the Internet, to its legitimate users. This is achieved either by exploiting the vulnerabilities in the software, network protocols, or operation systems, or by exhausting the consumable resources such as the bandwidth, computational time and memory of the victim. The first kind of attacks can be avoided by patching-up vulnerable software and updating the host systems from time to time. In comparison, the second kind of DoS attacks are much more difficult to defend. This works by sending a large number of packets to the target, so that some critical resources of the victim are exhausted and the victim can no longer communicate with other users.

In the distributed form of DoS attacks (called DDoS), the attacker first takes control of a large number of vulnerable hosts on the internet, and then uses them to simultaneously send a huge flood of packets to the victim, exhausting all of its resources. There are a large number of exploitable machines on the internet, which have weak security measures, for attackers to launch DDoS attacks, so that such attacks can be executed by an attacker with limited resources against the large, sophisticated sites. The attackers in DDoS attacks always modify the source addresses in the attack packets to hide their identity, and making it difficult to distinguish such packets from those sent by legitimate users. This idea, called IP address spoofing has been used in major DDoS attacks in the recent past, including the attacks on e-commerce

giants like *Yahoo*, *Amazon*, *Microsoft*, and *eBay*.

These recent DDoS attack used highly sophisticated and automated tools which ironically are readily available over the Internet, to be downloaded and used by anyone, even computer novices, to attack any Web site. Network worms have been developed and are available for the automatic scanning, exploitation, deployment, and propagation process of the attack tools.

The devastating effects of the DoS and DDoS attacks have caused attention of scientists and researchers, leading to various mechanisms that have been proposed to deal with them. However, most of them are ineffective against massively distributed DoS attacks involving thousands of compromised machines. In this paper, we present and analyze a Marking-based Detection and Filtering (MDADF) scheme to defend massively distributed DoS attacks. The rest of the paper is organized as follows. In Section 2 we discuss the existing approaches for defending DDoS attacks and argue why they are not adequate. In Section 3 we state the objectives of our proposed solution and then introduce the idea behind our scheme in Section 4. The complete scheme is described in detail in Section 5 followed by an analysis of the experimental results in Section 6. In Section 7 we compare our scheme with an existing marking-based scheme (called the *Pi* scheme), to illustrate the improvements achieved by our scheme. Finally, Section 8 concludes the discussion.

2 Approaches for Defending DoS/DDoS Attacks

Current DoS/DDoS defenses can be classified into three categories: preventive mechanisms, reactive mechanisms, and source-tracking mechanisms.

2.1 Preventive Defence

The preventive schemes aim at improving the security level of a computer system or network; thus preventing the attacks from happening, or enhancing the resistance to attacks.

A *proactive server roaming* scheme [15] belongs to this category. This system is composed of several distributed homogeneous servers and the location of active server changes among them using a secure roaming algorithm. Only the legitimate users will know the server's roaming time and the address of new server. All connections are dropped when the server roams, so that the legitimate users can get services at least in the beginning of each roaming epoch before the attacker finds the active server out again.

Such solutions are generally costly and difficult to really prevent attacks.

2.2 Source Tracking

The source-tracking schemes, on the other hand, aim to track-down the sources of attacks, so that punitive action can be taken against them and further attacks can be avoided. The existing solutions fall into four groups: packet marking, message traceback, logging, and traffic observation.

Many different *packet marking* schemes have been proposed, for encoding path information inside IP packets, as they are routed through the internet. The idea is first put forward by Savage et al. [21], called probabilistic packet marking (PPM), in which the routers insert path information into the Identification field of IP header in each packet with certain probability, such that the victim can reconstruct the attack path using these markings and thus track down the sources of offending packets. Song and Perrig improve PPM basing on a preestablished map of upstream routers, and provide authentication to the markings by encoding them using MAC functions [24]. Dean et al. [11] mention an algebraic approach based on reconstructing polynomial functions to track packets. Peng et al. [20] propose to reduce the number of packets needed for the attack path reconstruction in PPM, by dynamically changing the marking probability of a router according to its location in the path. If each router marks packets with a fixed probability, the victim needs to wait for the packets marked by the routers farther away from it, which are relatively fewer. Therefore, the farther a router is to the victim, the higher the marking probability should be. Belenky and Ansari [2, 3] propose a deterministic marking approach (DPM), in which only the address of the first ingress interface a packet enters instead of the full path the packet passes (as used in PPM) is encoded into the packet.

In the *message traceback* method [4, 17], routers generate ICMP traceback messages for some of received packets and send with them. By combining the ICMP packets with their TTL differences, the attack path can be determined. Some factors are considered to evaluate the value of an ICMP message, such as how far is the router to the destination, how quick the packet is received after the beginning of attack, and whether the destination wishes to receive it.

Another method called *logging* [22, 23, 26] is to record packet information at routers. The path to the attacker can be determined by the routers exchanging information with each other.

The *traffic-observation* method [5] is to determine the attack path by observing the rate change of attack traffic. During an attack, basing on the knowledge of the Internet topology, the victim floods an incoming link with excessively large numbers of packets, so that the attack traffic will be reduced if it comes from this link. By performing the link test recursively, the attacker can be finally found out.

A common problem existing in these four solutions is that the reconstruction of attack path becomes quite com-

plex and expensive when there are a large number of attackers (i.e. for highly distributed DoS attacks). Also, these types of solutions are designed to take corrective action after an attack has happened and cannot be used to stop an ongoing DDoS attack.

2.3 Reactive Solutions

The reactive measures for DDoS defence are designed to detect an ongoing attack and react to it by controlling the flow of attack packets to mitigate the effects of the attack.

One of the proposed reactive schemes, given by Yaar et al. [27] uses the idea of packet marking for filtering out the attack packets instead of trying to find the source of such packets. This scheme uses a path identifier (called *Pi*) to mark the packets; the *Pi* field in the packet is separated into several sections and each router inserts its marking to one of these. Once the victim has known the marking corresponding to attack packets, it can filter out all such packets coming through the same path.

The *Pushback* [14] method generates an attack signature after detecting a congestion, and applies a rate limit on corresponding incoming traffic. This information is then propagated to upstream routers, and the routers help to drop such packets, so that the attack flow can be pushed back.

D-WARD [18] is designed to be deployed at the source network. It monitors the traffic between the internal network and outside and looks for the communication difficulties by comparing with predefined normal models. A rate-limit will be imposed on any suspicious outgoing flow according to its offensive.

A *PacketScore* [16] scheme estimates the legitimacy of packets and computes scores for them by comparing their attributes with the normal traffic. Packets are filtered at attack time basing on the score distribution and congestion level of the victim.

In the *Neighbor Stranger Discrimination (NSD)* [1] approach, NSD routers perform signing and filtering functions besides routing. It divides the whole network into neighbors and strangers. If the packets from a network reach the NSD router directly without passing through other NSD routers, this network is a neighbor network. Two NSD routers are neighbor routers to each other if the packets sending between them do not transit other NSD routers. Therefore, a packet received by an NSD router must either from a neighbor networks, or from a neighbor router. Each NSD router keeps an IP addresses list of its neighbor networks and a signatures list of its neighbor routers. If a packet satisfies neither of the two conditions, it is looked as illegitimate and dropped.

The success of the reactive schemes depends on a precise differentiation between good and attack packets.

3 Designing an Effective Protection Scheme

Generalizing from the various defense mechanisms, a good protection scheme against DDoS attacks should be based on continuous monitoring, precise detection and timely reaction to attacks. The following characteristics are desirable:

- The scheme should be able to control or stop the flow of attack packets before it can overwhelm the victim. The timely detection and immediate reaction to an attack is essential, to prevent the depletion of resources at the victim location. The suitable place to deploy defense scheme are the perimeter routers or the firewall of a network.

- In stopping the flow of attack packets to the victim, the scheme must ensure that packets from legitimate users are successfully received so that the service to the legitimate users is not denied or degraded. Any degradation in service would signify a partial success for the denial of service attack.

- The implementation cost should be low. Unless most internet users fully recognize the threats posed by DoS/DDoS attacks, it is difficult to get cooperation from them in defending such attacks, especially when the investment required is costly. Therefore, any viable DDoS defence scheme should require minimal participation of third party networks or intermediate routers on the internet.

A good defence mechanism should be able to precisely distinguish the attack packets from the legitimate packets. What makes it difficult to control or stop the DDoS attacks is the use of spoofed IP address.

Spoofed packets are commonly used in DoS/DDoS attacks to hide the location of attackers and the compromised machines, so that the paths to them are concealed. Also, the success of the reflector attacks and many of the basic DoS attacks require the use of spoofed IP addresses in the attack packets [7]. In the reflector attack, attackers flood the victim through some hosts called reflectors. They control the compromised hosts to send a large number of packets to many reflectors with spoofed source IP addresses of the victim. All the reflectors will send responds to the victim, so that the effect of the attack is amplified many times. Also, the attack path becomes unclear due to the participation of reflectors. Some of the DoS attacks, such as smurf, fraggle, land, and the flood attacks, need to spoof their packets, using the victim's or random IP address, to fulfill their attacks.

If we can distinguish the packets which have spoofed IP addresses, then these packets can be selectively filtered out by a firewall to stop most attacks.

4 Distinguishing the Attack Packets

In this section, we present our packet marking method which will help us to distinguish DDoS attack packets from packets sent by legitimate users.

Though source IP addresses can be spoofed by attackers, the paths packets take to the destination are totally decided by the network topology and routers in the Internet, which are not controllable by the attackers. Therefore, the path of a packet has taken can really show the source of it. By recording the path information, the packets from different sources can be precisely differentiated, no matter what the IP addresses appeared in the packets. Packet marking, which is firstly proposed by Savage et al. in the PPM scheme [21], is a good method to record path information into packets.

To indicate the path a packet traverses, the simplest way is to add all the routers' IP addresses into the packet. The number of hops a packet passes through in the Internet is about 15 on average and mostly less than 31 [9, 10]. Since the length of a path is uncertain, it is difficult to reserve enough space in the packet to put all the addresses, and the packet size increases as the length of the path increases.

In order to avoid the increase in packet size, a possible method is to put all information into a fixed space. A router puts its IP address into the marking space of each packet it receives; if there is already a number in that space, it calculates the exclusive-or (XOR) of its address with the previous value in the marking space and puts the new value back. This method ensures that the marking does not change its length when a packet travels over the Internet, so the packet size remains constant.

In order to make the marking scheme fast and efficient we use part of the header in an IP packet, as the marking field. The 16-bit Identification field in IP header has been commonly employed as the marking space (see [2, 3, 20, 21, 24, 27]). The Identification(ID) field is currently used to indicate IP fragments belonging to different packets, but only less than 0.25% of the packets on the Internet actually use this feature [25]. Therefore, employment of ID-field as the marking space will not much affect the normal transmission of IP packets.

The PPM scheme [21] used packet marking to traceback to the attack sources. However, traceback becomes quite inefficient when the attackers amount increases. Moreover, finding out the sources of attack packets can only stop these compromised machines from sending more attack packets, but usually cannot discover the genuine attackers hiding behind them. Therefore, a better idea of defending should be to identify the attack flows and stop them from reaching the victim.

In our scheme each cooperating router on the path of an IP packet would insert a mark on the ID-field of the packet. The generated marking should be such that two packets reaching the victim through different routes are guaranteed to have distinct markings.

4.1 Computing the Packet Marking

The mark made by a router would be a function of its IP address. To fit the 32-bit IP address A of a router into the ID field, we employ a hash function h that converts A to a 16-bit value. We adopt the CRC-16 hash function which is easy to compute and has low collision rate.

Since attackers can easily know the routers' IP addresses, they can spoof the marking on a packet if they know the hash function used by each router. We cannot expect every router in the Internet to participate in the marking scheme and mark all packets passing through it. If a packet with such a spoofed marking passes through a route where there are no co-operating routers, this packet is impossible to be identified as an attack packet.

To avoid such spoofing of the marking, each router R uses a 16-bit key K_R (which is a random number chosen by the router) when computing its marking. The marking for a router R is calculated as $M_R = \mathrm{h(A)}$ XOR K_R, where A is the IP address of the router. After receiving a packet the router computes the marking $M = M_R \oplus M_{old}$, if an old marking M_{old} exists in that packet, and replaces M_{old} with M.

4.2 Inserting Order Information

One possible drawback with the scheme mentioned above is that the marking on a packet depends only on the routers it passes through, but not on the order passing them. This means that the packets which pass the same routers on two different paths have the same marking.

To make the marking scheme more effective, we let each router perform a Cyclic Shift Left(CSL) operation on the old marking M_{old} and compute the new marking as $M = CSL(M_{old}) \oplus M_R$. In this way, the order of routers influences the final marking on a packet received by the firewall.

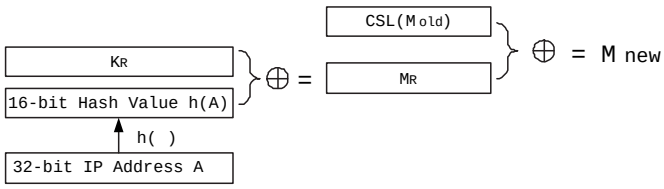

Figure 1: The marking scheme

The complete marking scheme is shown in Figure 1, and the pseudo code is described below:

```
Marking procedure at router R (having IP address A):

    k <- a 16-bit random number
    M(R) <- k XOR h(A)
    For each packet w
    {
```

```
        If W.ID = 0 Then
            w.ID <- M(R)
        Else
        {
            M_old <- w.ID
            M_new <- M(R) XOR CSL(M_old)
            w.ID <- M_new
        }
    }
```

5 Filtering Scheme

The MDADF scheme employs a firewall at each of the perimeter routers of the network to be protected and the firewall scans the marking field of all incoming packets to selectively filter-out the attack packets (see Figure 2).

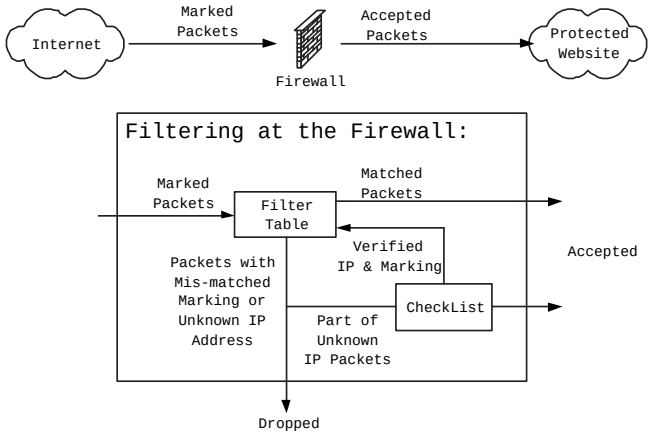

Figure 2: The system structure

On employing our marking scheme, when a packet arrives at its destination, its marking depends only on the path it has traversed. If the source IP address of a packet is spoofed, this packet must have a marking that is different from that of a genuine packet coming from the same address. The spoofed packets can thus be easily identified and dropped by the filter, while the legitimate packets containing the correct markings are accepted.

5.1 Learning Phase

To distinguish the spoofed packets, the firewall needs to keep a record of the genuine markings. During normal time that no attacks are happening, the firewall can learn about the correct markings for packets sent from specific IP addresses. The (IP-address, Marking) pairs are stored in a *Filter Table*[1], which are later used to verify each incoming packet and filter-out the spoofed ones. The learning phase continues for a sufficient time to allow most of the filter table to be filled up. If the Filter Table gets full, any new entry to be added replaces the oldest one.

[1]The filter table can be implemented as a content-addressable memory to speed up the filtering process.

5.2 Normal Filtering Procedure

After the learning phase, the firewall begins to perform its normal filtering operations. To the packet from an IP address recorded in the Filter Table, it is accepted if it has a consistent marking; otherwise, it is dropped. For the packet from a new IP address, we accept it with probability p and put the (IP-address, Marking) pair to a *Check List*, so that the marking can be verified. The value of p is set to high (close to 1) initially. When an attack is detected, the value of p is decreased according to the packet arrival rate and the victim's capability for handling the incoming traffic.

5.3 Marking Verification

To verify the markings in the Check-List, a random *echo* message is sent periodically to the source address for each (IP-address, Marking) pair in the Check-List, and a counter is used to record the number of echo messages have been sent for it. To avoid the reply being imitated by the attacker, the content of the echo message is recorded in the Check-List and compared with the content of reply received.

On receiving an echo reply from the source, the marking can be verified and the (IP-address, Marking) pair is moved to the Filter Table; otherwise, it indicates the previously received packet was spoofed, then this pair is deleted from the Check List. If the counter in the Check List shows that more than $d(= 10)$ echo messages have been sent to an IP address x, then the entry for this IP address is removed from the Check List and the pair (x,ϕ) is added to the filter table, where ϕ is a special symbol denoting that all packets having source IP address x should be discarded. Since in this situation, this source IP must be either non-existent or inactive, so that the packets received with this source address are coming from the attacker and need to be rejected.

5.4 Attack Detection

To detect the start of a DDoS attack, we use a counter called Total-Mismatches-Counter (TMC), which counts the number of packets whose marking cannot be matched at the firewall. This includes both packets with incorrect markings as well as packets from unknown source addresses that are not recorded in the Filter Table. When the TMC value becomes greater than a threshold θ, it is considered as a signal of DoS/DDoS attack. The value of TMC is reset to zero after fixed intervals to ensure that the cumulative results over a long duration is not considered as the indication of attack by mistake.

5.5 Route Change Consideration

Though routes on the Internet are relatively stable, they are not invariable. Once the route between two hosts has changed, the packet received by the destination will have a different marking with the one stored in the Filter Table,

so that it may be dropped according to our basic filtering scheme.

Taking route changes into consideration, we introduce another counter called SMC, to count the number of mismatching packets for any IP address A. When the value of SMC_A reaches a threshold δ, the entry (A, $Marking_A$) is copied to the Check List to test whether the route from this source has changed and SMC_A is reset to zero. If the new marking is verified by the Check List verification process, the marking for this IP address is updated in the Filter Table. Otherwise, the original marking is preserved. Unless the route change has been verified, the original marking is still used to filter packets.

5.6 Complete Filtering Scheme

Using the techniques and criteria introduced above, a complete filtering procedure is described below. Any packet received by the firewall is judged by the filter according to the following rules:

1) If the (IP-address, Marking) pair is same with one of the records in the Filter Table, the packet is received.

2) If the source IP address of the packet exists in the Filter Table, but the marking does not match, this packet is considered to be a spoofed packet and is dropped. TMC is incremented.

3) If the source IP address does not appear in the Filter Table, then this packet is accepted with a probability p. TMC is incremented.

4) If the TMC value exceeds the threshold, an attack is signaled.

5) All echo reply messages that are received as responses to the firewall's requests are handled by the Check List verification process. They are not passed through the filter.

In general, our MDADF scheme has the following functions:

- Distinguish and filter out spoofed packets by checking the marking of each packet using the Filter Table.
- Detect the occurrence of DDoS attack, so that appropriate defensive measures can be taken before serious damage is caused.
- Ensure that not many legitimate packets are dropped mistakenly, due to route changes on the Internet.

5.7 Pushback Implementation

By employing the filtering scheme, the firewall can protect the victim Web site by filtering out attack packets. However, sometimes the attack flow may be too large and the firewall may not have enough resources to handle it. In that case, we may employ the method of pushback [14].

In the Pushback method, the victim of a DDoS attack sends the signatures of attack to upstream routers and ask them to help filtering out these packets. Since one IP address can be used in the attack packets from many different sources, if we use the markings of spoofed packets as the attack signatures, large numbers of comparison need be done by the upstream routers. Instead, we create a list of IP addresses with their corresponding markings from the Filter Table and send this list (called the Pushback List) to the upstream routers.

Whenever the firewall adds new entries or updates old entries in the Filter Table, these entries are sent as updates to the upstream routers, so that the Pushback List can be updated. The upstream routers compare each packet with the Pushback List after marking it and discard spoofed packets. Most of the attack packets are filtered before arriving at the victim, so that the victim Web site can continue with its normal operations.

In some instances, the upstream routers of the victim still cannot deal with the attack flow, then they need to pushback further. To perform this function, each router R transforms all original markings $M_i(i = 0, 1, \cdots, n)$ in the Pushback List by computing $M_i' = CSR(M_i \oplus M_R)$, where CSR (Cyclic Shift Right) is the inverse of the CSL operation. The router then sends the new generated markings $M_i'(i = 0, 1, \cdots, n)$ to its upstream routers. This process can be performed recursively until the attack flow is controlled.

6 Experimental Results

We have evaluated the performance of the MDADF scheme under various parameter settings by simulating DDoS attacks of different magnitudes.

6.1 Simulation of Internet Traffic

In our simulation, we have used the topological data obtained from the Internet Mapping Project [9] of Lumeta Corporation. This data was generated by using *traceroute* to probe the paths in Internet from a single host (netmapper.research.lumeta.com, 65.198.68.56). Since all the paths in the database congregate at the single node, this node is quite suitable to act as the victim in the simulations of DDoS attacks.

We use a packet generator process to simulate the normal Internet traffic, which periodically sends packets from a randomly selected internet user. Then the packet marking process is simulated, by computing the markings for each cooperating router on the route for this particular user. Finally, the marked packet is inserted into a packet-queue at the firewall of the victim. The rate, at which packets are added to the packet-queue, mimics the normal traffic flow for a typical Web-server on the Internet.

Attackers usually have two methods to disguise the source locations: spoofing a genuine host's IP address, or inserting a randomly generated IP address into source

address field. We simulated both types of attacks, called *Spoofed* attack and *Randomized* attack respectively. Packets are generated from each attacker to simulate the attack traffic. So, higher the number of attackers, more will be the volume of the attack flow. In the simulation of Spoofed attack, for each attack packet, one of the legitimate user is randomly selected and its IP address is used as the spoofed value of the source address. The marking field is initially filled with a random value and the marking process is simulated, as before.

6.2 Parameter Selection

The choice of values for different parameters affects the performance results of the MDADF scheme. In our experiments, we have come up with some suitable values for these parameters by trail and error and we have tested the effects of changing the values of these parameters. The data-set used in the experiments contained 10,000 hosts and 50,000 intermediate routers. The size of the filter table was varied from 5000 to 10,000. The participation rate of routers was varied from 100% to 0%. For the parameter p, the most suitable value were found to 0.75 and 0.1 respectively for the pre-attack and post-attack scenarios. A learning phase of 10 minutes gave good results in our simulation setting. In the following, we discuss the results obtained during the experiments.

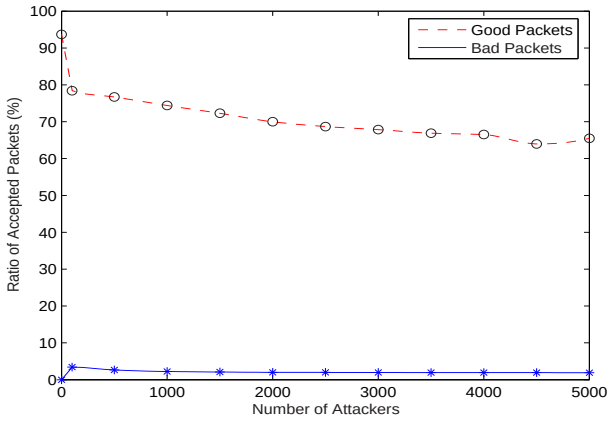

Figure 3: Ratio of accepted packets vs. different amount of attackers under spoofed attack

6.3 Performance under Spoofed Attack

Figure 3 shows the ratio of packets that were accepted at the firewall under different magnitudes of attack. As can be seen, more than 70% good packet are accepted even in the most severe condition under spoofed attack, in which the attack traffic is almost 10 times of the normal traffic. As the number of attackers is increased, there is a slight decrease in the acceptance rate of legitimate packets, as due to the heavy congestion some packets are dropped before the firewall can handle them. Though the good packets acceptance ratio decreases a little with the increasing number of attackers, the bad packets accepting ratio stays at a very low level.

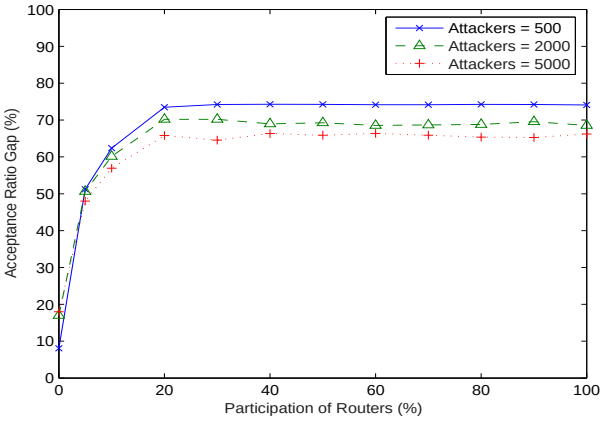

Figure 4: Acceptance ratio gap vs. different participation percentage under spoofed attack

The participation of routers in the marking schene is important to our MDADF scheme; However we cannot expect all the routers to be willing to cooperate. Therefore, we have tested the effect of different participation rates on our scheme when the attackers are 500, 2000, and 5000 respectively. We show the difference between acceptance ratio of good and bad packets, which is called "Acceptance Ratio Gap". Obviously, if no markings are applied, the acceptance ratio of good and bad packets should be equal and the gap is zero. As shown in Figure 4, many more good packets are accepted than the bad ones even when the participation rate is only 20% under all three conditions. This means that the MDADF scheme can efficiently distinguish between good and bad packets when just 20% of routers in the Internet deploy our marking scheme.

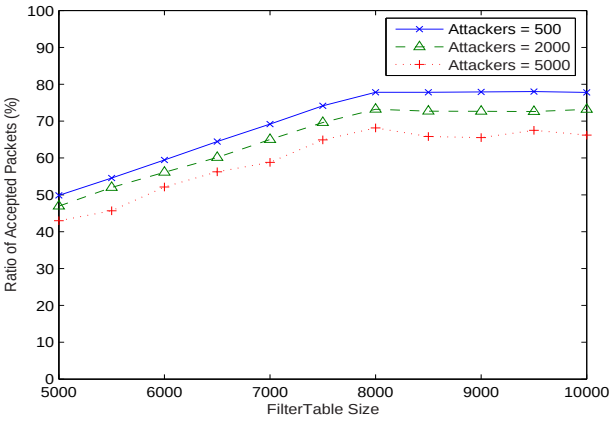

Figure 5: Acceptance ratio gap vs. different filter table size under spoofed attack

The Filter Table is a critical part of our system, which

stores the (IP-address, Marking) pairs and more the number of records in the Filter Table, less good packets will be dropped by mistake. We have tested the performance of our scheme with different sizes of Filter Table under different attack environments and Figure 5 shows that keeping 80% of the legitimate user records is sufficient to filter packets, and even 70% is good enough.

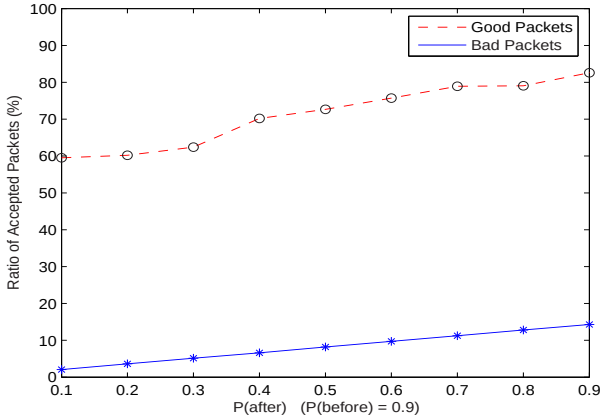

Figure 6: Ratio of accepted packets vs. different probability value under spoofed attack

The value of probability p affects the acceptance ratio of both good and bad packets whose IP addresses are not recorded in the Filter Table. Figure 6 exhibits the influence of the value of p (after detection) on the acceptance ratio of packets, when the initial value of p (before the attacks) is set at 0.9.

Though relatively more good packets are accepted with the increase in the value of p, the the percentage of attack packets that are accepted also increases (which, in real-life, could lead to more packets being deleted from the queue).

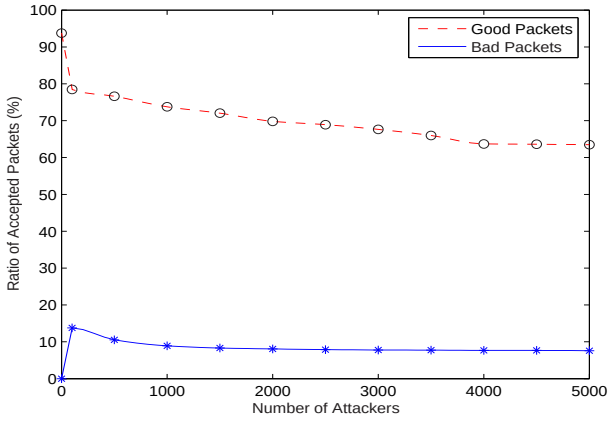

Figure 7: Ratio of accepted packets vs. different number of attackers under randomized attack

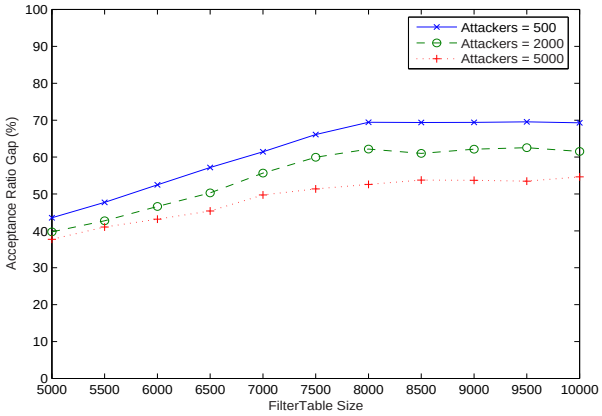

Figure 8: Acceptance ratio gap vs. different filter table size under randomized attack

6.4 Performance under Randomized Attack

We tested our scheme's performance under randomized attack in which attackers use randomly generated IP addresses. Figures 7, 8, and 9 show the variations in the packet acceptance ratio on changing the magnitude of attacks, using various Filter Table size, or using different values for p respectively. The results show that our scheme is effective even under attacks of high magnitude. The acceptance ratio of bad packets increases rapidly with the value of p, because most attack packets contain IP addresses that are new to the system.

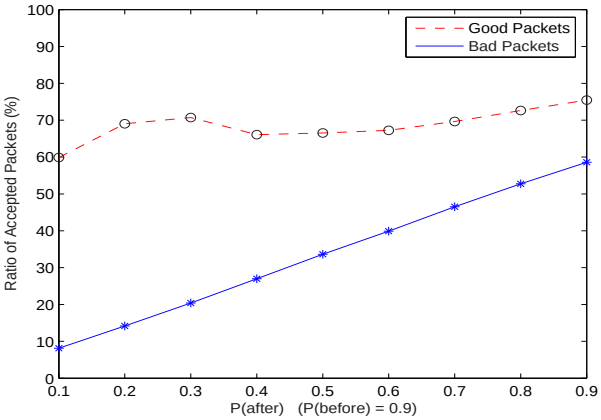

Figure 9: Ratio of accepted packets vs. different probability value under randomized attack

6.5 Attack Detection Time

Under our simulation setting, the MDADF scheme detected the occurrence of an attack in 3 - 4 seconds in most conditions as shown in Table 1. The results were almost same for the spoofed attacks and the randomized attacks. When the number of attackers is too small (100 or less), it takes a little more time for the system to notice the effects of the attack.

Table 1: Attack detection time for different number of attackers

# of Attackers	Attack Detection Time (sec)	
	Spoofed Attack	Randomized Attack
100	6.86	6.85
500	3.66	3.63
1000	3.57	3.56
1500	3.54	3.51
2000	3.54	3.51
2500	3.54	3.51
3000	3.51	3.50
3500	3.51	3.48
4000	3.50	3.48
4500	3.51	3.49
5000	3.51	3.49

7 Comparison with the *Pi* Scheme

The Path Identification (Pi) mechanism proposed in [27] is similar with our MDADF scheme, in that they are both deterministic and filter packets basing on packet marking. The Pi scheme can identify attack packets coming from a same path after knowing the first one. The identification is basing on the marking, Path Identifier, of each packet, which is same to all packets passing through the same path. Though the marking does not include the information of entire path an attack packet has traversed, as other traceback mechanisms do, it can help the victim to distinguish attack packets basing on the markings they have.

7.1 Marking Scheme

In the Pi scheme, the ID field of each packet is divided into $\lfloor 16/n \rfloor$ parts, and each router inserts n bits of marking into the packet. Since the last several bits of IP addresses are usually clustered at a few numbers, such as 0 and 1, then the markings of different routers cannot be distinguished in this way. To avoid the iteration, a router computes the MD5 hash value of its IP address, which makes the value of its last n bits address distributed. Furthermore, to the packets coming from different upstream routers, a router uses different markings to distinguish them. Since the space is limited, not all markings of routers along the path can be put into, so that the later routers will overwrite the markings of previous routers. To ensure the packets from one origination will have the same marking and will not be affected by the attacker's manipulation, the marking is adjusted every time so that the oldest marking always appears in a fixed location. To keep the markings of different routers in a packet as many as possible, n can be 1 or 2 and the internal nodes of autonomous systems can omit marking, because the internal routes can be known from the local administration department.

In the MDADF scheme, all the routers on the path of a packet contribute to the marking. Each router computes a 16-bit hash value of its IP address, with a secret 16-bit value, to construct its marking. Every time the old marking of a packet XORs with the router's marking to form the new marking.

7.2 Filtering Scheme

To filter packets, the Pi scheme needs to be informed about the attack packet first, learns its marking, then begins to block packets with the same marking. Therefore, it needs to store the attack markings. When the number of attackers is large, lots of markings will be blocked, then it is quite possible for legitimate packets to have the same markings and be dropped. Therefore, to decrease the false positive, a threshold can be applied so that a packet will be dropped only when the percentage of attack packets having the same marking exceeds the threshold.

In the MDADF scheme, the genuine (IP-address, marking) pairs are stored. The packets with mis-matching markings are considered as spoofed packets from attackers and dropped. For the packets with source IP addresses not kept in the archives, they are accepted with certain probability. Furthermore, the markings of packets with unknown IP addresses will be verified through a marking verification process, as well as those stored IP addresses from which many packets with different markings have been received.

Both of the two schemes filter packets basing on the marking of packets. They do not use the marking to find the source of attack, but use it to separate attack packets. Then, the victim do not need to wait until enough packets have been received to reconstruct the path as the traceback mechanisms do. Therefore, these two mechanisms have low overhead to both the routers and the victim. Both mechanisms can be combined with pushback.

The Pi scheme filters packets basing on known attack markings. A disadvantage of it is that, if an attacker spoofs the marking or inserts random value and the marking is not overwritten by other routers, the victim cannot recognize the attack packet. While the MDADF filtering scheme use genuine markings of IP addresses, the packets with mis-matching markings are dropped. Even if

no router marks the attack packet, the genuine marking cannot be spoofed by attackers in our marking scheme, because each router's marking is a function of its own secret key, which is not available to attackers.

The Pi scheme cannot distinguish between legitimate and attack packets by itself, unless it knows the markings of attack packets from an external source. Therefore, this source must be reliable and capable to provide precise information. Accordingly, the MDADF scheme differentiates attack packets automatically by using the records of genuine (IP-address, marking) pairs.

Route change is considered in the MDADF scheme. Since the path between two hosts in the Internet is not invariable, when we use recorded markings to differentiate packets, good packets will be classified falsely if their routes have changed. To reduce the false positive caused by the change, echo messages are used to test the genuine markings marked by routers at current time.

Though a threshold is used in packet filtering process of Pi scheme, the false positive and false negative are inevitable and determined by the threshold value. If the value is low, many legitimate packets will be dropped; otherwise, lots of attack packets will come through, yet higher threshold is more suitable when the number of attackers is large. Again, the threshold is the percentage of attack packets versus the total number of packets having the same marking. In practical implementation, it is difficult to know a packet is actually an attack packet or legitimate one, which is the aim of most DDoS defense schemes. Therefore, it is hard to say a packet is from attacker while another having the same marking is from legitimate user. Moreover, if the victim can really get the percentage, he would have already had the ability to precisely differentiate attack packets, then he does not need to recur to the threshold to filter packets and lead to good packets be dropped mistakenly.

7.3 Performance Comparison

The Pi scheme can accept at most 60% more good packets than bad ones when 100% routers participate the marking, while the advantage decreases to as low as 5% when the participation is 50%. In MDADF scheme, we get an acceptance ratio gap of 70% until the participation rate decreases to 20% and the gap is still bigger than 60% when only 10% routers cooperate (as in in Figure 4). The much less participation requirement to the Internet routers indicates much less implementation overhead required by MDADF. Moreover, our MDADF scheme has the functions to detect the occurrence of DDoS attacks automatically and distinguish attack packets. The Pi scheme cannot do the filtering unless it is told which are the attack packets and has learned the attack markings.

The mainly differences between PI and MDADF schemes are listed in Table 2.

8 Conclusions and Discussion

In this paper, we have proposed a low-cost and efficient scheme called MDADF, for defending against DDoS attacks, The MDADF scheme is composed of two parts: marking process and filtering process. The marking process requires the participation of routers in the Internet to encode path information into packets. We suggest the use of a hash function and secret key to reduce collisions among packet-markings. The scheme also includes mechanisms for detecting and reporting DDoS in a timely manner.

The evaluation of the scheme under simulations, show that our scheme can effectively and efficiently differentiate between good and bad packets under spoofed attack when the routers' participation rate is as low as 20%, so the deployment cost of our scheme is very low. Also, most good packets are accepted even under the most severe attack, whose traffic is about 10 times of normal traffic. At the same time, the bad packet acceptance ratio is maintained at a low level. Our scheme performs well even under massively distributed DoS attacks involving upto 5000 attackers.

Under both spoofed and randomized DDoS attacks, the MDADF scheme detected the occurrence of attack precisely within 3 - 4 seconds. The quick detection is valuable to the victim so that appropriate actions can be taken to minimize the damage caused by a DDoS attack.

Acknowledgements

The authors would like to thank Bill Cheswick and Hal Burch for providing the internet dataset used in our simulations.

References

[1] N. Aaaraj, S. Itani, and D. Abdelahad, "Neighbor stranger discrimination (NSD)- A new defense mechanism against DDoS attacks," in *Proceedings of the 3rd FEA Student Conference*, May 2004.

[2] A. Belenky and N. Ansari, "IP traceback with deterministic packet marking," *IEEE Communications Letters*, vol. 7, no. 4, pp. 162-164, Apr. 2003.

[3] A. Belenky and N. Ansari, "Tracing multiple attackers with deterministic packet marking (DPM)," in *2003 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing (PACRIM'03)*, pp. 49-52, Aug. 2003.

[4] S. Bellovin, *ICMP Traceback Messages*, Internet draft, work in progress, Mar. 2000.

[5] H. Burch and B, Cheswick, "Tracing anonymous packets to their approximate source," in *Proceedings of the 14th Systems Administration Conference (LISA'00)*, pp. 319-327, Dec. 2000.

Table 2: Comparison between PI and MDADF

	PI	MDADF
Marking scheme	Each router inserts marking into a section of the ID field. Later markings overwrite previous ones.	XORing of each router's marking.
Filtering scheme	Based on the markings of known attack packets.	Based on the markings of legitimate users.
Distinguishing Attack Packets	Must be informed by an external source at the first time, then records its marking to filter.	Automatically, by using the Filter Table.
Attack Detection Functionality	None	Yes
Acceptance Ratio Gap	60% when all routers participate, and decreases to 5% when the participation rate is 50%.	70% when all routers participate, and holds even if the participation rate decreases to 20%.

[6] CERT® Coordination Center, *CERT/CC Statistics 1988-2005*. http://www.cert.org/stats/cert_stats.html#incidents.

[7] Y. Chen, *A Novel Marking-based Detection and Filtering Scheme Against Distributed Denial of Service Attack*, Masters Thesis, University of Ottawa, 2006.

[8] Y. Chen, S. Das, P. Dhar, A. E. Saddik, and A. Nayak, "An effective defence mechanism against massively distributed denial of service attacks," in *the 9th World Conference on Integrated Design & Process Technology (IDPT'06)*, San Diego, June 2006.

[9] B. Cheswick and H. Burch, *Internet Mapping Project*, http://research.lumeta.com/ches/map/.

[10] Cooperative Association for Internet Data Analysis, Skitter, 2000. (http://www.caida.org/tools/measurement/skitter/)

[11] D. Dean, M. Franklin, and A. Stubblefield, "An algebraic approach to IP trackback," in *Proceedings of the 2001 Network and Distributed System Security Symposioum*, pp. 3-12, Feb. 2001.

[12] Internet System Consortium, *ISC Domain Survey: Number of Internet Hosts*, http://www.isc.org/index.pl?/ops/ds/host-count-history.php.

[13] Internet World Stats, *Internet User Statistics – The Big Picture: World Internet Users and Population Stats*, http://www.internetworldstats.com/stats.htm

[14] J. Ioannidis and S. M. Bellovin, "Implementing pushback: router-based defense against DDoS attacks," in *Proceedings of the Network and Distributed System Security Symposium (NDSS'02)*, pp. 6-8, Feb. 2002.

[15] S. M. Khattab, C. Sangpachatanaruk, R. Melhem, D. Mosse, and T. Znati, "Proactive server roaming for mitigating denial-of-service attacks," in *Proceedings of the 1st International Conference on International Technology: Research and Education (ITRE'03)*, pp. 500-504, Aug. 2003.

[16] Y. Kim, W. C. Lau, M. C. Chuah, and H. J. Chao, "PacketScore: statistics-based overload control against distributed denial-of-service attacks," in *Proceedings of IEEE INFOCOM'04*, pp. 2594-2604, Mar. 2004.

[17] A. Mankin, D. Massey, C.-L. Wu, S. F. Wu, and L. Zhang, "On design and evaluation of "Intention-driven" ICMP traceback," in *IEEE International Conference on Computer Communication and Networks (ICCCN'01)*, pp. 159-165, Oct. 2001.

[18] J. Mirkovic, G. Prier, and P. Reiher, "Attacking DDoS at the source,"in *Proceedings of the IEEE International Conference on Network Protocols*, pp. 312-321, Nov. 2002.

[19] K. Park and H. Lee, "On the effectiveness of route-based packet filtering for distributed DoS attack prevention in power-law internets," in *Proceedings of ACM SIGCOMM'01*, Aug. 2001.

[20] T. Peng, C. Leckie, and K. Ramamohanarao, "Adjusted probabilistic packet marking for IP traceback," in *Networking 2002*, pp. 697-708, May 2002.

[21] S. Savage, D. Wetherall, A. Karlin, and T. Anderson, "Practical network support for IP traceback," in *Proceedings of ACM SIGCOMM'00*, Aug. 2000.

[22] A. C. Snoeren, C. Partridge, L. A. Sanchez, C. E. Jones, F. Tchakountio, S. T. Kent, and W. T. Strayer, "Hash-based IP traceback," in *Proceedings of ACM SIGCOMM'01*, pp. 3-14, Aug. 2001.

[23] A. C. Snoeren, C. Partridge, L. A. Sanchez, C. E. Jones, F. Tchakountio, B. Schwartz, S. T. Kent, and W. T. Strayer, "Single-packet IP traceback," *IEEE/ACM Transactions on Networking*, vol. 10, no. 6, pp. 721-734, Dec. 2002.

[24] D. X. Song and A. Perrig, "Advanced and authenticated marking schemes for IP traceback," in *Proceedings of IEEE INFOCOM'01*, pp. 878-886, Apr. 2001.

[25] I. Stoica and H. Zhang, "Providing guaranteed services without per flow management," in *Proceedings of ACM SIGCOMM'99*, pp. 81-94, Apr. 1999.

[26] R. Stone, "CenterTrack: An IP overlay network for tracking DoS floods," in *Proceedings of USENIX Security Symposium*, pp. 199-212, Aug. 2000.

[27] A. Yaar, A. Perrig, and D. Song, "Pi: A path identification mechanism to defend against DDoS attacks," in *Proceedings of the IEEE Symposium on Security and Privacy*, pp. 93-109, May 2003.

Yao Chen received her B.Eng. degree in Computer Science from Zhengzhou University, China, in 1999 and the Master degree in Computer Science from the University of Ottawa, Canada, in 2006. Her current research interests include Internet security, DDoS attack detection and defense, and Internet traffic analysis.

Shantanu Das is currently a Ph.D. candidate at the School of Information Technology and Engineering, University of Ottawa, Canada. He received his M.Tech and BIT degrees in computer science from the Indian Statistical Institute and the University of Delhi, in 2004 and 2002, respectively. His current research interests include distributed computing, fault tolerance in distributed systems and network security.

Pulak Dhar received his B.Tech degree in Electronics and Electrical Communication Engineering from the Indian Institute of Technology in 1969. He was an engineer in the Semiconductors Division of Bharat Electronics Ltd., Bangalore, India, from 1969 till 1982 working in the area of development, production and quality assurance of semiconductor devices, integrated circuits and hybrid microcircuits. In January 1983 he joined Northern Telecom Ltd in Canada as a component quality and reliability assurance engineer and was later given the responsibility as a senior engineer for software quality and reliability assurance. Since 2001 he is with Cistel Technology Inc as R&D Manager, managing research and development projects on software engineering, wireless communication and security systems for the information technology sector.

Abdulmotaleb El Saddik (IEEE M'02-SM'03) is an associate professor at the School of Information Technology and Engineering (SITE) at the University of Ottawa. He is the director of the Multimedia Communications Research Laboratory (MCRLab) and of the Information Technology Cluster, Ontario Research Network on Electronic Commerce. He has authored and co-authored two books and more than 100 publications in the areas of knowledge management, development of multimedia artefacts and collaborative haptic virtual environments. Dr. El Saddik is a Distinguished IEEE Lecture. He is Editor of the International Journal of Advanced Media and Communication and Associate Editor of the ACM Journal of Educational Resources in Computing (JERIC). He serves in the program committee (as chair) of several IEEE conferences and workshops related to multimedia communications and Haptics. He was the recipient of the "Premier's Research Excellence Awards" (PREA round 10).

Amiya Nayak received his B.Math. degree in Computer Science and Combinatorics & Optimization from University of Waterloo in 1981, and Ph.D. in Systems and Computer Engineering from Carleton University in 1991. He has over 17 years of industrial experience, working at CMC Electronics (formerly known as Canadian Marconi Company), Defence Research Establishment Ottawa (DREO), EER Systems and Nortel Networks, in software engineering, avionics and navigation systems, simulation and system level performance analysis. He has been an Adjunct Research Professor in the School of Computer Science at Carleton University since 1994. He had been the Book Review and Canadian Editor of VLSI Design from 1996 till 2002. He is in the Editorial Board of International Journal of Parallel, Emergent and Distributed Systems, International Journal of Computers, Information Technology & Engineering, and the Associate Editor of International Journal of Computing and Information Science. Currently, Dr. Nayak is a Full Professor at the School of Information Technology and Engineering (SITE) at the University of Ottawa. His research interests are in the area of fault tolerance, distributed systems/algorithms, and mobile ad hoc networks with over 90 publications in refereed journals and conference proceedings.