Hindawi Publishing Corporation
EURASIP Journal on Wireless Communications and Networking
Volume 2006, Article ID 96157, Pages 1–9
DOI 10.1155/WCN/2006/96157

ZSBT: A Novel Algorithm for Tracing DoS Attackers in MANETs

Xin Jin,[1] Yaoxue Zhang,[1] Yi Pan,[2] and Yuezhi Zhou[1]

[1] *Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China*
[2] *Department of Computer Science, Georgia State University, University Plaza, Atlanta, GA 30303, USA*

Received 24 August 2005; Revised 15 March 2006; Accepted 3 April 2006

Denial of service (DoS) attack is a major class of security threats today. They consume resources of remote hosts or network and make them deny or degrade services for legitimate users. Compared with traditional Internet, the resources, such as bandwidth, memory, and battery power, of each node are more limited in mobile ad hoc networks (MANETs). Therefore, nodes in MANETs are more vulnerable to DoS attacks. Moreover, attackers in MANETs cannot only use IP spoofing to conceal their real identities but also move arbitrarily, which makes it a challenging task to trace a remote attacker in MANETs. In this paper, we proposed a zone sampling-based traceback (ZSBT) algorithm for tracing DoS attackers in MANETs. In our algorithm, when a node forwards a packet, the node writes its zone ID into the packet with a probability. After receiving these packets, the victim can reconstruct the path between the attacker and itself. Simulations were carried out to illustrate the validity of the algorithm; even with a little communication overhead.

1. INTRODUCTION

A MANET is a collection of mobile nodes that establish communication paths dynamically. Nodes may join a network at any time and communicate with the entire network via neighboring nodes. In recent years, with the rapid deployment of MANET applications, securities become one of the major problems in MANET today. MANETs are much more vulnerable to various kinds of attacks [1] than wired networks due to their characteristics, such as the volatile network topologies, dependence on collective participation of all nodes, and the limited bandwidth and battery power of nodes.

Attacks against MANETs can be classified into two categories: passive attacks and active attacks. Passive attacks typically involve eavesdropping of data. Active attacks involve actions such as replication, modification, and deletion of exchanged data or DoS attacks. This kind of attacks always target at congestion, propagating incorrect routing information, preventing services from working properly, or stopping them completely.

DoS attacks by an unintentional failure or malicious action are one of the major classes of threats in network security today. A classical way of DoS attack is to flood any centralized resources to make them no longer operate correctly or even crash. In MANET, besides the classical way of DoS attack, a more concealed form used in an open MANET environment is the so-called sleep deprivation torture. In this type of DoS attack, the attacker is trying to deprive a device with limited battery power by sending a large number of legal packets to the victim to keep it awake and engaged in the communication all the time. The neighbor nodes of the attacker are difficult to detect this type of attack by their own intrusion detection system, because both the behavior of the attacker and the packets it sent are legal. The victim itself may detect the attack very quickly because it can find that a large number of packets have no actual operations or the operations do not make sense.

When a victim detects a DoS attack, a widely used solution is tracing the DoS attack back towards its origin, and then stopping the attacker at the source. As attackers usually use IP spoofing to conceal their real location, several IP traceback mechanisms have been proposed for the Internet, such as link testing [2], ingress filtering [3], probabilistic packet marking (PPM) [4], and ICMP traceback (ITrace) [5], to trace the true sources of attackers. These traceback approaches cannot be directly applied to MANET due to the following reasons that are related to two aspects: efficiency and effectivity.

(1) Nodes in MANETs can move arbitrarily, which makes attack paths change frequently. Therefore, additional constraints are placed on tracing approaches for locating the attack sources in time. Therefore, the traceback approaches

used in MANETS should be more effective than that in the Internet.

(2) Traceback approaches in the Internet always consume a lot of bandwidth, computational resources, and battery power. However, in MANETs, nodes are typically devices with limited bandwidth, computational resources, and battery power. These limitations require that the traceback approaches in MANETs should be more efficient than that in the Internet.

Concentrating on how to effectively and efficiently trace remote DoS attackers in MANET environment, we proposed a zone sampling-based traceback (ZSBT) algorithm. In ZSBT, the network area is divided into several zones and each node knows its zone ID. When a node receives a packet to be forwarded, it first writes its zone ID with a probability p into the packet and then forwards the packet. When it detects that it is suffered from a DoS attack, the victim can reconstruct the entire path by combining a modest number of such packets. We study the performance of ZSBT algorithm using GloMoSim [6] simulator with different marking probability. The simulation results have shown the validity of ZSBT.

The rest of the paper is organized as follows. In Section 2, we discuss the related work. In Section 3, details of the ZSBT algorithm are presented. In Section 4, we give the performance analysis. Simulation model and simulation results are provided in Section 5. Section 6 concludes this paper.

2. RELATED WORK

Savage and his colleagues have proposed a probabilistic packet marking (PPM) approach to reconstruct the path from a remote attacker to the victim in the Internet [4]. The basic idea behind PPM is the usage of edge sampling. A packet on the path is marked with a certain probability by two routers on the way, forming an edge. Each marked packet then represents a sample of the whole path. The victim receives all packets and can thereby use the marked packet to reconstruct the entire path back to the source. The number of data packets, X, required for the victim to reconstruct an attack path of d hops, has the following bounded expectation:

$$E(x) < \frac{\ln(d)}{p(1-p)^{d-1}}. \qquad (1)$$

However, this approach needs additional 72- bit space in the IP packet header, as we all know that there is no so much space in the IP packet header. What we can use is only the 16-bit identification field, so the author proposed an encoding approach to compress the 72- bit information into 16 bits. But the encoding approach needs a mass of computation, which is not efficient for the portable devices.

ICMP traceback (ITrace) was first proposed by Bellovin and his colleagues [5]. The basic idea behind ITrace is that every router should sample a packet with a small probability, copy its content onto a special ICMP packet, add information about the adjacent upstream and/or downstream routers, and send it towards the same destination as the original packet. The victim of an attack can then use these packets to reconstruct the paths back to the attackers. An enhancement to ITrace, known as ITrace-CP (ICMP traceback with cumulative path) [7], was proposed, thereby the ITrace-CP messages are made to carry the entire attack path information so as to facilitate a faster attack path construction in the event of DoS attacks. When a router receives an IP packet, an ITrace-CP message will be generated based on the probability set by the router. This message is then sent to the next hop router, instead of the destination address of the IP packet. In [8], Vrizlynn et al. have proposed an enhanced ITrace-CP to trace attackers in both wired networks and wireless ad hoc networks. In their approach, they consider distribution of the probability in an exponential manner so that a faster construction time is achievable within the same overhead constraint. As the PPM approach requires overloading a field in the IP header, which raises the backward protocol compatibility problem, ITrace/ITrace-CP utilizes out-of-band messaging to achieve the packet tracing purpose. The shortcomings of this approach are the following: first, it will bring some additional bandwidth consumption; second, due to the unpredictable routing topology, the packet loss ratio in MANET is much larger than that in the Internet; therefore it will need more ICMP packets to guarantee the victim to receive enough ICMP packets.

In [9], Kim and Helmy have proposed a small world-based attacker traceback (SWAT) approach to trace DoS attacker in MANET. They use traffic patterns matching (TPM) and traffic volume matching (TVM) as matching-in-depth techniques to identify DoS attackers. And then, to efficiently search relay nodes on the attack path, they extend small world-based contact model [10] and propose a (multi-) directional search method for DoS/DDoS attacker traceback using contact nodes, which can reduce communication overhead in energy constrained MANETs and increase traceback robustness against collusion of partial nodes. Note that this approach is an on-demand approach, that is, when the victim detects DoS attack, it begins to broadcast query packets. However, firstly, on-demand approaches first consume additional bandwidth and batter power; and secondly, it will take a longer time to find out the attacker. When the attacker information has been transmitted back to the victim, it is possible that the attacker has already moved to other places [10].

3. ZSBT ALGORITHM FOR MANETS

3.1. *Differences between Internet and MANET when tracing a DoS attacker*

To trace a remote DoS attacker in MANET is an extremely challenging task. Two main reasons are as the following. First, an attacker can spoof a source address, which results that the victim cannot figure out who is the real attacker only through the source address. Second, the topology of MANET always changes, so the packets from the attacker to the victim may change to different paths several times over a short period. However, the only invariant that can be depended on is that a packet from the attacker must traverse all the nodes along the path between it and the victim. Therefore, if each packet can record some path information, when the victim receives enough packets, it can reconstruct the path using the information in those packets. Then the remaining problem is that what information should be recorded and how to record the information in each packet. To solve the problem, the edge sampling method is used in the PPM approach, which can effectively trace a remote attacker in the Internet.

Enlightened by the PPM approach, the ZSBT algorithm is proposed in this paper, which can trace the remote DoS attacker effectively and efficiently in MANET environments. Firstly, we will introduce the differences between Internet and MANET when tracing a DoS attacker.

(1) In the Internet, DoS attackers and the victims are always not in the same subnet. The packets sent by the attacker first need to be transmitted to the gateway and then transmitted by the routers on the path, and finally arrive at the victim. The gateway is a computer or router which has a fixed IP address. Therefore, the goal of tracing a DoS attacker in the Internet is to find out the subnet where the attacker belongs. MANET is used mostly in some special situation temporarily. The nodes in MANET can move arbitrarily; therefore, the relative position between two nodes may change frequently. Therefore, there is not a fixed gateway for each node. Consequently, the addresses of nodes are always flat addresses. Even using IP address, they are in the same subnet. In this situation, tracing the DoS attacker in MANET is not to find out the attacker's subnet like that in the Internet but the physical position area.

(2) In the Internet, if the attacker's subnet has been found out, the attacker is difficult to displace itself to another subnet in a short time. And the paths that the packets have passed through are not changed frequently. In MANET, however, the paths which the packets have passed through are changed frequently; thus the needed time for tracing the attacker should be very short; otherwise the attacker may move to another position before the tracing process is completed.

(3) In the Internet, routers, switches, and PCs have strong computational abilities, unlimited battery power, and 100 M bandwidth. The tracing algorithm can be more complex and therefore more accurate. However, in MANET, the portable devices have no such advantaged resources and then the tracing algorithm should be rather simple than accurate.

3.2. Reasons for sampling zone

Firstly, two notions are defined. Node path is a path between the source and destination composed by nodes through which the data flow passes. Zone path is a path between the source and destination composed by zones through which the data flow passes.

In the ZSBT algorithm, a network area is divided into several zones. The creation and the maintenance of zones are beyond the research topic of this paper. The partitioning of the network could be based on the simple geographic partitioning or other clustering algorithms [9]. We assume that the zone partitioning mechanism is accurate and safe. One simple approach to obtain the zones is based on geographic partitioning. With the help of GPS, it is possible that a mobile host knows its physical location. Then the node can determine its zone ID by mapping its physical location to a zone map. When a packet passes through a node, the node writes its zone ID instead of its IP address into the packet, as that in the PPM approach, mainly for the following reasons.

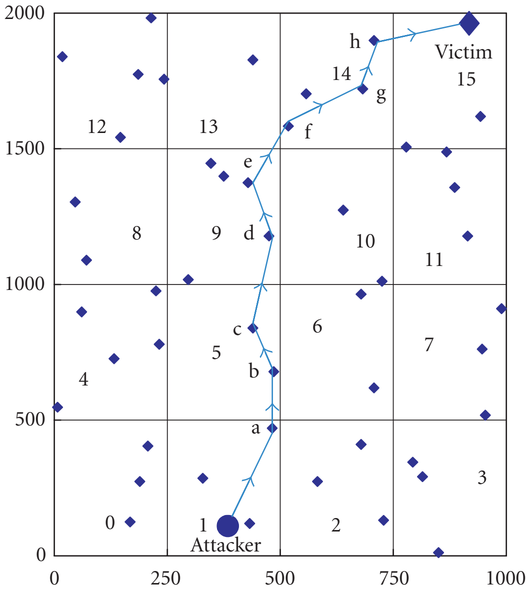


Figure 1: Node path versus zone path (node path = 9 hops).

(1) Using the zone, the path length can be restricted in a relatively small value. For example, in Figure 1, the node path between the attacker and the victim can be reconstructed through 9 hops. However, the zone path is through only 5 hops. If the node path between the attacker and the victim has extended to 15 hops, the zone path is sill through 5 hops as in Figure 2.

(2) Node path may change frequently due to the mobility of nodes, but the zone where a node stays will be changed more slowly; thus the zone path is steadier than the node path. Moreover, once the zone where the attacker stays has been found out, it can be considered that in most cases the attacker cannot leave the zone instantly.

(3) To record IP address, a packet needs to reserve at least 4 bytes. In the PPM approach, if the edge sampling method is used, the packet needs to reserve 9 bytes to record 2 IP addresses and one distance field. However, to record zone ID, 1 byte can represent 256 different zones. This saves a lot of space in the IP packet header.

3.3. ZSBT algorithm

The ZSBT algorithm consists of three processes: initialization process, zone sampling process, and path reconstruction process. The flow chart of ZSBT algorithm is shown in Figure 3.

Step 1. Initialization process. In the initialization process, each node constructs a chain and lets the victim be the head.

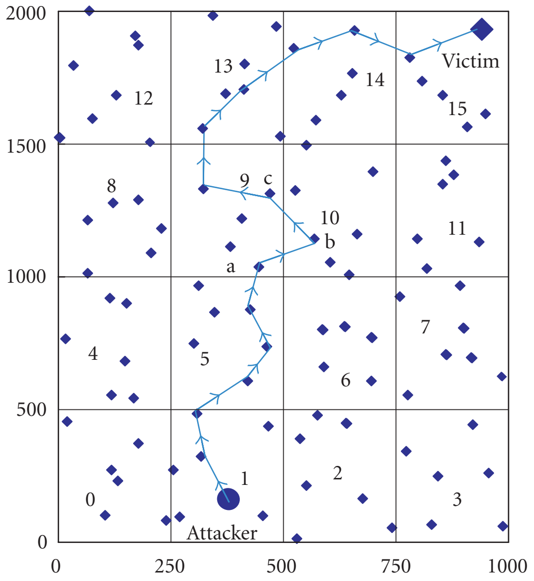


Figure 2: Node path versus zone path (node path = 15 hops).

The chain is used to reconstruct the attack path by sorting the zone ID information in the packets.

When a node receives a packet, if the node is the victim, the ZSBT algorithm goes to Step 3; the *path reconstruction process* is executed. Otherwise, the ZSBT algorithm goes to Step 2, the *zone sampling process* is executed.

Step 2. Zone sampling process. In the zone sampling process, the node writes its zone ID into the node with a probability p and then forwards the packet. Two static fields, zone *ID*, and *distance* in each packet are reserved. zone *ID* is used to record the zone ID of the node on the path. *Distance* represents the distance from current node to the victim and its initial value is set as zero. The concrete actions each node takes are as the following.

(a) Get its zone ID from the zone map. The method to divide zones and to get zone ID has been discussed above.

(b) Engender a random number x from [0,1) and compare it with the marking probability p.

(c) If $x < p$, then the node writes its zone ID into the zone *ID* field and writes 1 into the *distance* field in the packet, and then forwards the packet.

(d) Otherwise, if the zone *ID* field is not null, then the node compares its zone ID with the value in the zone *ID* field in the packet. If they are equal, the packet will be forwarded directly, otherwise, the *distance* field will be increased by 1 and then the packet is forwarded.

The zone sampling process is described in Algorithm 1.

Step 3. Path reconstruction process. In the path reconstruction process, the victim reconstructs the zone path from the attacker to itself using the zone information in each packet. The detailed steps are as the following.

(a) Insert the value of zone *ID* in the received packet into the chain according to the value of *distance*.

(b) If the value of zone *ID* in the packet is equal to the value of zone *ID* in the chain, then the old value is replaced by the new value.

The path reconstruction process is described in Algorithm 2.

If the chain is constructed successfully, the victim can then find out all the zones that the packet has been passed through. Then the attack response methods can be used. There are some routing protocols in the MANET that use multiple paths to transmit packets. If using this kind of routing protocols, only one path is constructed because the victim can launch certain methods to prevent the attack if only the victim can trace back to the zone where the attacker stays using one zone path.

Here, it is needed to point out that packets do not sample the edge between two ordinal zones in the ZSBT algorithm as in the PPM. The reason is as follows. In the edge sampling method, packets record the IP address of the nodes at each end of a link, when the victim wants to insert a packet into the path tree, it can compare the *start* field in the packet with the *end* field of the nodes in the path tree. If the *start* field in the packet is equal to the *end* field of one node, it means that the packet should be inserted right after this node. But in the ZSBT algorithm, the path changes all the time. Thus, even two ordinal zones are recorded; the *start* field may be not equal to the *end* field of any node in the path chain. Therefore, only the *distance* field is used to sort the zone ID.

3.4. A brief example

Figure 4 is a brief application of the ZSBT algorithm. The points represent the nodes, the arrows between two nodes represent the path that the packets have passed through, and the numbers in this figure represent the zone IDs. The Attacker is in zone 1. It is assumed that the attacker is launching a DoS attack to the victim through the nodes b->c->d->e->f->g->h->i->j->victim.

Under the above circumstance, each node firstly constructs a chain and lets itself be the head. When receiving a packet, node *b* can decide that it is not the destination from the packet header. Thus, zone sampling process is executed in the node b. The node b maps its coordinate into the zone map and gets its zone ID 2. Then the node b writes its zone ID into the zone *ID* field in the packet with a probability p. If the node b decides to mark the packet, it writes its zone ID into the zone *ID* field and sets the *distance* field as 1. If not, it compares the value of zone *ID* field in the packet with its own zone ID. If they are not equal, it increases the *distance* field by 1. After that, the node b forwards the packet. The continuous nodes along the path take the same actions as that of the node b.

When the victim receives this packet with the sampling zone $ID = 2$ and $distance = 4$, it can first decide it is the destination. Then, the path reconstruction process is executed. The victim itself inserts the value of zone *ID* into a chain

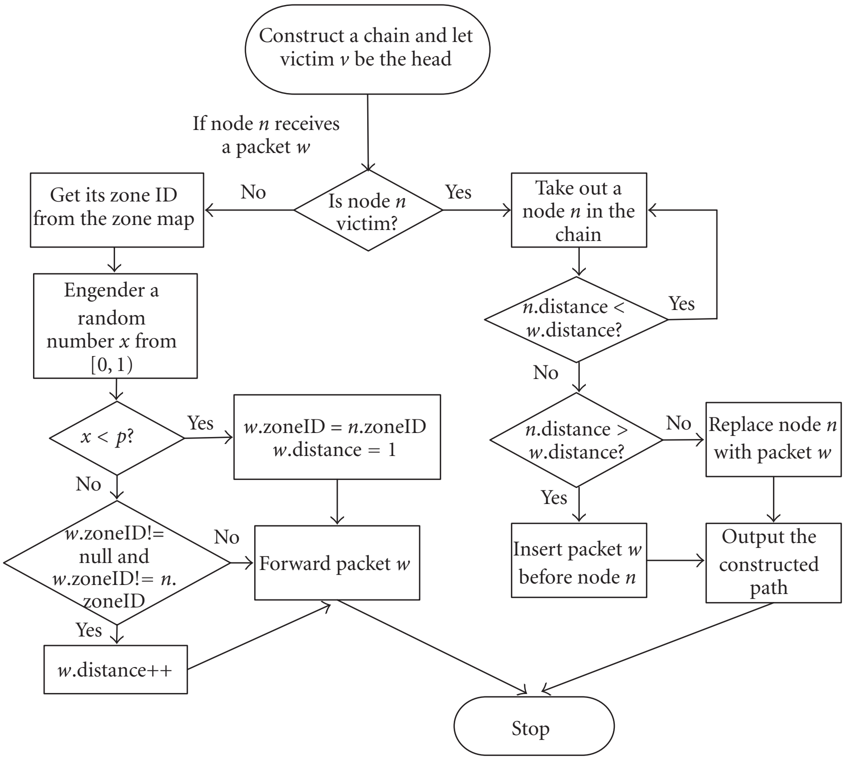

Construct a chain and let victim v be the head
If node n receives a packet w
Is node n victim?
No
Yes
Get its zone ID from the zone map
Take out a node n in the chain
Engender a random number x from $[0, 1)$
n.distance < w.distance?
Yes
No
$x < p$?
Yes
w.zoneID = n.zoneID w.distance = 1
n.distance > w.distance?
No
Replace node n with packet w
No
Yes
w.zoneID!= null and w.zoneID!= n.zoneID
No
Forward packet w
Insert packet w before node n
Output the constructed path
Yes
w.distance++
Stop

Figure 3: Flow chart of the ZSBT algorithm.

according to the value of *distance*. After receiving enough of such packets, the victim can reconstruct a zone path between the attacker and itself. In this example, the zone path is 5->4->3->2->1.

4. PERFORMANCE ANALYSIS

In the following section, we will discuss how many packets the victim needs to reconstruct a D hop zone path. In an area whose length is X and width is Y, if it is divided into zones whose length is x and width is y, then the number of the zones is $(X \cdot Y)/(x \cdot y)$. Let L be the longest distance that a packet passes through in the zone, then

$$L \leq \sqrt{x^2 + y^2}. \tag{2}$$

The radio range of nodes is the function of the radio transmission power. Under the same transmission power, different propagation models will produce different radio ranges. Let tx be the transmission power and l the radio range, then $l = f(tx)$.

Let n be the number of nodes that will forward the packet when a packet passes through some zone. Based on (2), n can be approximately computed as

$$n \approx \frac{L}{l} \leq \frac{\sqrt{x^2 + y^2}}{f(tx)}. \tag{3}$$

```
Marking procedure at node n:
    for each packet w{
        let x be a random number from [0, 1)
        if (x < p)   {
            write n.ZoneID to w.zoneID;
            w.distance = 1;
        }
        else {
            if ((w.zoneID != null)&&(w.zoneID != n.zoneID))
            w.distance++;
        }
    }
    forward packet w;
```

Algorithm 1: Zone sampling process.

Because every node marks the packet with probability p, the probability for the victim to receive a packet marked by a d hop away zone is

$$p(d) = (1 - (1 - p)^n)\left[(1 - p)^n\right]^{d-1} \quad (0 < d \leq D). \tag{4}$$

Because the probability of receiving a sample decreases geometrically as it is the further away from the victim, the convergence time for this algorithm is dominated by the time to receive a sample from the furthest route. Then the

```
Path reconstruction procedure at victim v:
    let v be the head of chain c;
    for each packet w from attacker {
        for each node n in the chain {
            if (w.distance == n.distance)
                                replace n with w;
            else insert w.zoneID into c according to w.distance
        }
    }
```

Algorithm 2: Path reconstruction process.

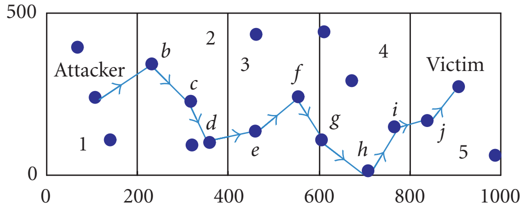


Figure 4: An example of the ZSBT algorithm.

expectation of the time can be expressed as

$$E(t) = \frac{1}{(1-(1-p)^n)\left[(1-p)^n\right]^{D-1}}. \qquad (5)$$

For convenient computing, it is conservatively assumed that samples from all of the D nodes appear with the same likelihood as the furthest node. From the point of the victim, when it receives a packet, the probability that the packet has some zone information is larger than

$$p(i) = D\left[1-(1-p)^n\right]\left[(1-p)^n\right]^{D-1}. \qquad (6)$$

From the well-known coupon collector problem, then the expected number of trials required to select one of each of D equiprobable items is

$$E(n) = D\big(\ln(D) + O(1)\big). \qquad (7)$$

Therefore, the number of packets required for the victim to reconstruct a zone path of length D has the following bounded expectation:

$$\begin{aligned} E(X) &= \frac{E(n)}{P(i)} \\ &< \frac{\ln(D)}{\left[1-(1-p)^{\sqrt{x^2+y^2}/f(tx)}\right]\left[(1-p)^{\sqrt{x^2+y^2}/f(tx)}\right]^{D-1}}. \end{aligned} \qquad (8)$$

From (8), we can discover that the value of $E(x)$ has close correlation with the value of p. Assume the function of p is as the following:

$$f(p) = \left[1-(1-p)^{\sqrt{x^2+y^2}/f(tx)}\right]\left[(1-p)^{\sqrt{x^2+y^2}/f(tx)}\right]^{D-1}. \qquad (9)$$

$f(p)$ is an incremental function of p, so $f(p)$ gets its maximal value when $\partial f(p)/\partial p = 0$, and at the same time $E(x)$ can get its minimal value. Therefore we can calculate the value of p

$$p = 1 - \sqrt[\sqrt{x^2+y^2}/f(tx)]{1-\frac{1}{D}}. \qquad (10)$$

5. SIMULATIONS

5.1. Simulation environment

We implemented ZSBT algorithm using the GloMoSim [5] library. The GloMoSim library is a scalable simulation environment for wireless network systems, especially for MANETs. It is designed as a set of library modules, each of which simulates a specific wireless communication protocol in the protocol stack. The library has been developed using PARSEC, a C-based parallel simulation language. Our simulation models a network within a rectangular region. Compared with a square region, the rectangular region can enlarge the average path length; so we can observe the performance on a longer path. One border of the region is 1000 meters, and we can change path length by changing the other border length. In most experiments unless specified, the network consists of 100 nodes and the mobility model is random waypoint model (pause time 30 s, min speed 5 m/s, max speed 10 m/s). The nodes in the network are placed uniformly. Radio transmission power is 10 dBm, and the propagation model is TWO-RAY. The packet size is 512 K byte, and the packet sending rate of DoS attacker is 100 packets per second. We run each scenario three times and the data collected are averaged over those runs.

5.2. Simulation results

First, we compare the number of zones with the length of zone path. In the simulation, the network area is divided into $X \times Y$ zones ($X = 4$, $Y = 2, 3, 4, 5, 6$). For each kind of zone division, two nodes whose distance is the longest are selected. As shown in Figure 5, with the increment of zone number, the length of zone path is also increasing, but the increasing rate is slow. When the number of zones varies from 8 to 24, the length of zone path only varies from 4 to 9. Thus, in a MANET with large area, we can increase the number of zones to obtain the attacker's position more accurately. Also, the zone path length increases slowly.

The length of zone path is related to the value of X and Y. Under the same zone number, if $X = 1$, $Y = 8, 12, 16, 20, 24$, the length of zone path must increase. Therefore, when dividing zones, we should make X be equal to Y.

In Figure 6, we compare the length of node path with the length of zone path when the network area is divided into 16 (4×4). Let the length of node path varies from 8 to 15

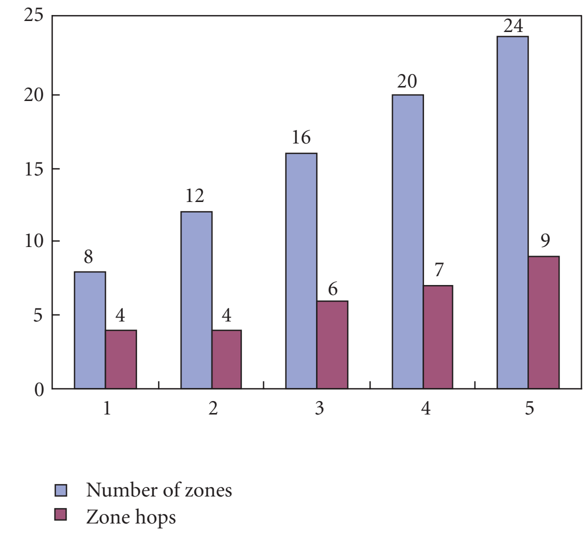


Figure 5: The comparison between the number of zones and the average length of the zone path.

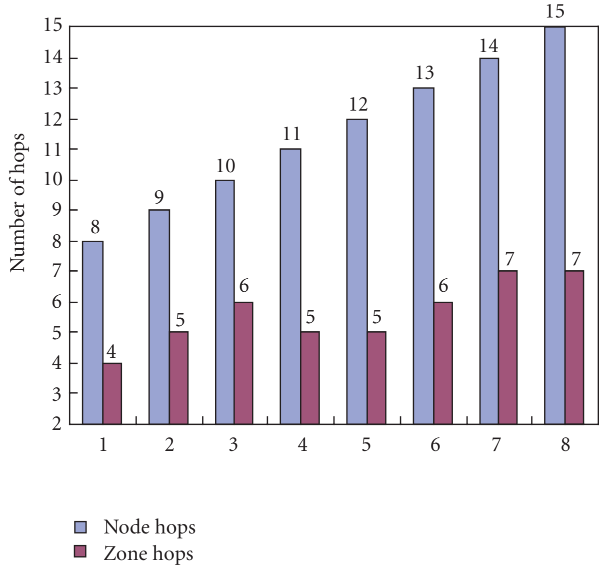


Figure 6: The comparison between the length of node path and the length of zone path.

hops, as shown in Figure 6, the zone path length only varies from 4 hops to 7 hops; and the length of zone hops is almost decided by the number of zones in the area. Therefore, the path length can be controlled as expected.

Figure 7 compares the number of packets to reconstruct a zone path between two nodes with different probabilities ($p = 0.2$ and $p = 0.05$). The distance between the two nodes varies from 8 to 15 hops. Because the length of the zone path is always no more than 7 hops, as shown in Figure 5, the number of packets to reconstruct the zone path is limited in a small number. From the figure, we can see that when the probability p is 0.05, the number of packets needed is no more than 50 packets. When the probability p is 0.2 the number of packets is no more than 40 packets. What is the optimal value of probability p? According to (8), the minimal value of $E(X)$ is gotten if p is adopted as $1 - \sqrt[\sqrt{x^2+y^2/f(tx)}]{1 - 1/D}$. Note that $\sqrt{x^2 + y^2/f(tx)}$ is approximately equal to 2 under our simulation parameters. In addition, the scope of the length of zone path D varies from 3 to 10 at most instances. Based on these two parameters, the probability p varies between 0.05 and 0.2. Thus in Figure 7, p is set as 0.05 and 0.2, respectively.

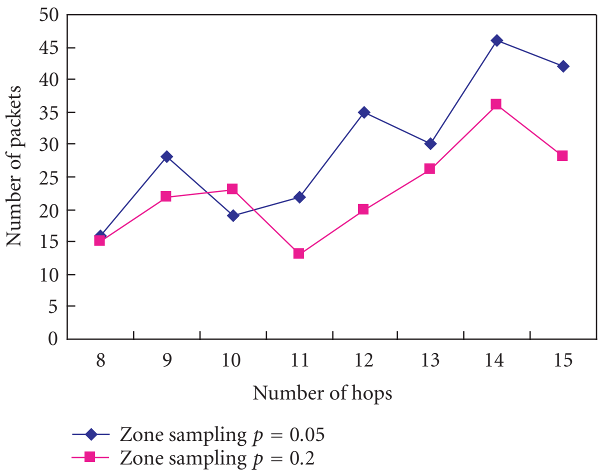


Figure 7: The number of packets needed to reconstruct the node paths with different lengths.

Figure 8 compares the theoretical value and the experimental value of the number of packets needed to reconstruct a path. The simulation environment of Figure 8 is as follows: 16 (4 × 4) zones, the area of each zone is 250 meters ×500 meters. When the radio transmission power is 10 dBm, and the propagation model is TWO-RAY, the radio transmission range is 282 meter. Figure 4 shows that if the network area is divided into 16 (4 × 4) zones, when the length of node path varies from 8 hops to 15 hops, the length of zone path varies from 4 to 7 hops. If these parameters are put into (8), it can be educed that the number of packets that the victim needs varies from 20 to 45 packets. The experimental values shown in Figure 6 varied from 8 hops to 15 hops which drop within the theoretical bound.

In the MANET, only if the attacker can be traced back before it moves away from the zone, the victim can launch certain methods to prevent the attack. Figure 9 shows the relationship between the settling time and the area of the zone. In the simulation, we choose the random waypoint model (pause time: 30 s, min speed: 5 m/s, max speed 10 m/s). One border length is fixed as 250 meters, and the other border length is 100, 200, 300, 400, 500 meters, respectively. Figure 9 shows that even in the smallest area, the node will stay for about 60 seconds. Figure 7 shows that the victim needs no more than 50 packets to reconstruct the path. To launch a DoS attack, the attacker at least needs to send dozens of

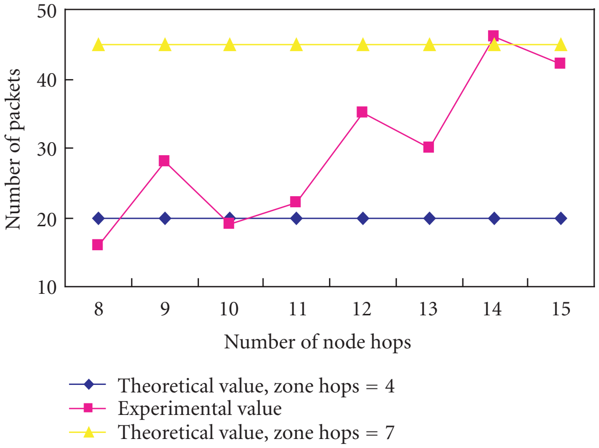


Figure 8: The comparison between theoretical value and experimental value of the number of packets needed to reconstruct a path.

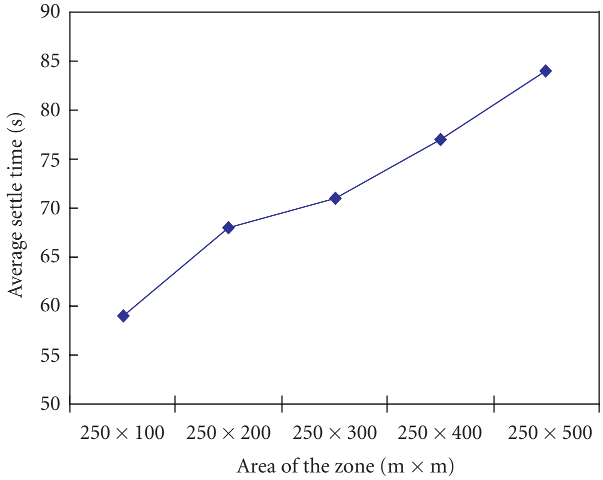


Figure 9: The relationship between average settle time and area of Zone.

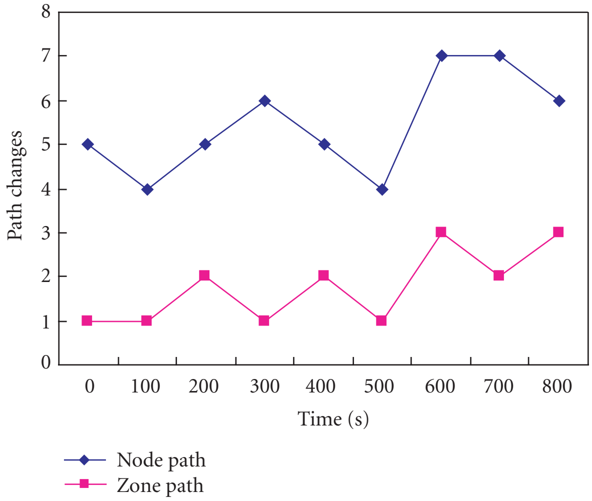


Figure 10: The comparison of times the node path and zone path are changed.

packets per second; thus the time needed to reconstruct the path is short enough before the attacker leaves its zone.

Figure 10 compares the times of the node and zone path changing within 100 seconds. We recorded the path change times every 100 seconds. From Figure 10, we can see that if the zone path is used, the path was changed about 2 times in 100 seconds. However, the node path was changed about 5 times in the same period. This shows that the change of the zone path is smaller than that of the node path, and it will provide a more advantageous ability to prevent DoS attack.

6. CONCLUSIONS

In this paper we have proposed a zone sampling-based traceback (ZSBT) algorithm used to trace DoS attacker in the MANET environment effectively and efficiently. ZBST algorithm uses the zone information of each node sampled by the packets to reconstruct the path between the attacker and the victim. In this algorithm, the convergence time is shorter and the per-packet space is smaller than other algorithms. Moreover, the accuracy of the attacker's position can be adjusted by changing the number of zones. The simulation results have demonstrated that this algorithm is capable of fully tracing most attacks after they send only a few decades of packets; then the victim can have enough time to take measures to prevent the attacks.

After the attacker has been traced, the victim can take several measures to prevent the attack. Here, we enumerate three measures. First, the victim can inform the zone path to which the nodes belong not to forward or reduce the priority of packets from the zone where the attacker stays. Second, if the position-based routing protocol is used in the network, the victim can send a routing error message to the nodes in the attacker's zone. Thus, the attacker will stop sending packets to the victim because it thinks that the victim is unreachable. Lastly, if there is an out-of-band communication method, the victim can inform the nodes in the attacker's zone that one of you has been compromised. Then the nodes in the attacker's zone will inspect themselves whether they are compromised, or will start up their own intrusion detection system to detect their neighbors.

However, there is a shortcoming of ZSBT algorithm. This scheme will sacrifice the accuracy of the path for tracing DoS attackers. One zone may include many nodes and the identification of hackers is not so precise. Although we have proposed several methods to prevent DoS attack in the above paragraph, the precision of ZSBT algorithm still needs to be improved.

In the future work, we will not only put our focus on locating the exact DoS attackers zone, but also extend our algorithm to trace DDoS attackers.

REFERENCES

[1] K. Wrona, "Distributed security: ad hoc networks & beyond," in *Proceedings of Ad Hoc Networks Security Pampas Workshop*, Rhul, London, UK, September 2002.

[2] R. Stone, "CenterTrack: an IP overlay network for tracking DoS floods," in *Proceedings of 9th USENIX Security Symposium*, pp. 199–212, Denver, Colo, USA, August 2000.

[3] P. Ferguson and D. Senie, Network Ingress Filtering: Defeating Denial of Service Attacks Which Employ IP Source Address Spoofing. RFC 2267, 1998.

[4] S. Savage, D. Wetherall, A. Karlin, and T. Anderson, "Practical network support for IP traceback," in *Proceedings of the ACM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication (SIGCOMM '00)*, pp. 295–306, Stockholm, Sweden, September 2000.

[5] S. Bellovin, M. Leech, and T. Taylor, "ICMP Traceback Messages," IETF Internet Draft, Version 4, February 2003.

[6] X. Zeng, R. Bagrodia, and M. Gerla, "GloMoSim: a library for parallel simulation of large-scale wireless networks," in *Proceedings of 12th Workshop on Parallel and Distributed Simulation (PADS '98)*, pp. 154–161, Banff, Alberta, Canada, May 1998.

[7] H. C. J. Lee, V. L. L. Thing, Y. Xu, and M. Ma, "ICMP traceback with cumulative path, an efficient solution for IP traceback," in *Proceedings of 5th International Conference on Information and Communications Security (ICICS '03)*, pp. 124–135, Huhehaote, China, October 2003.

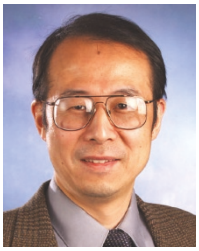
[8] V. L. L. Thing, H. C. J. Lee, M. Sloman, and J. Zhou, "Enhanced ICMP traceback with cumulative path," in *Proceedings of 61st IEEE Vehicular Technology Conference (VTC '05)*, vol. 4, pp. 2415–2419, Stockholm, Sweden, May-June 2005.

[9] Y. Kim and A. Helmy, "SWAT: small world-based attacker traceback in Ad-hoc networks," in *Proceedings of IEEE Infocom Poster/Demo Session (INFOCOM '05)*, Miami, Fla, USA, March 2005.

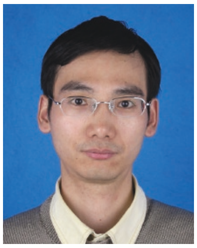
[10] A. Helmy, "Contact-extended zone-based transactions routing for energy-constrained wireless ad hoc networks," *IEEE Transactions on Vehicular Technology*, vol. 54, no. 1, pp. 307–319, 2005.

Xin Jin received his Bachelor's degree from the University of Science & Technology of China in 2001, and received his Master's and Ph.D. degrees in computer science from Tsinghua University, China, in 2006. Now he is a Researcher in China Mobile Communication Corporation Research Institute. Dr. Jin's research interests include routing protocols in ad hoc networks, security in wireless networks, and communication protocols in 3G core network.

Yaoxue Zhang is a Professor in the Department of Computer Science and Technology at Tsinghua University, China. He also serves as the Director General of the Higher Education Department, Ministry of Education (MOE), China. His research interests include computer network, operating systems, distributed computing system, and pervasive (ubiquitous) computing. He received his B. Eng. degree from Xidian University, China, in 1982, and his M.S. and Ph.D. degrees in engineering from Tohoku University, Japan, in 1989. He worked as a Visiting Scientist of the Institute of Computer Science at MIT in 1995.



Yi Pan was born in Jiangsu, China. He entered Tsinghua University in March 1978 with the highest college entrance examination score among all 1977 high school graduates in Jiangsu Province. Currently, he is the Chair and a Full Professor in the Department of Computer Science at Georgia State University. He received his B.Eng. and M.Eng. degrees in computer engineering from Tsinghua University, China, in 1982 and 1984, respectively, and his Ph.D. degree in computer science from the University of Pittsburgh, USA, in 1991. His research interests include parallel and distributed computing, optical networks, wireless networks, and bioinformatics. He has published more than 80 journal papers with 30 papers published in various IEEE journals. In addition, he has published over 100 papers in refereed conferences (including IPDPS, ICPP, ICDCS, INFOCOM, and GLOBECOM). He has also coedited 24 books (including proceedings) and contributed in several book chapters.



Yuezhi Zhou is an Associate Researcher at the Department of Computer Science & Technology at Tsinghua University, China. His area of research includes computer system architecture, network computing, and pervasive computing. Now his main research interest is to develop a new architecture for future service-oriented computing, named transparent computing, in which users can demand computing service in a hassle-free way.