A DEFENSE SYSTEM ON DDOS ATTACKS IN MOBILE AD HOC NETWORKS

Except where reference is made to the work of others, the work described in this dissertation is my own or was done in collaboration with my advisory committee. This dissertation does not include proprietary or classified information.

_____

Xuan Yu

Certificate of Approval:

_____	_____
Martin Carlisle Professor Department of Computer Science United States Air Force Academy	Drew Hamilton, Chair Associate Professor Computer Science and Software Engineering
_____	_____
Yu Wang Assistant Professor Computer Science and Software Engineering	Joe Pittman Interim Dean Graduate School

A DEFENSE SYSTEM ON DDOS ATTACKS IN MOBILE AD HOC NETWORKS

Xuan Yu

A Dissertation

Submitted to

the Graduate Faculty of

Auburn University

in Partial Fulfillment of the

Requirements for the

Degree of

Doctor of Philosophy

Auburn, Alabama
May 10th, 2007

A DEFENSE SYSTEM ON DDOS ATTACKS IN MOBILE AD HOC NETWORKS

Xuan Yu

_____
Signature of Author

_____
Date of Graduation

DISSERTATION ABSTRACT

A DEFENSE SYSTEM ON DDOS ATTACKS IN MOBILE AD HOC NETWORKS

Xuan Yu

Doctor of Philosophy, May 10th, 2007
(M.S., Auburn University, 2001)
(B.S., Zhejiang University, 1996)

180 Typed Pages

Directed by Drew Hamilton

Network security is a weak link in wired and wireless network systems. Malicious attacks have caused tremendous loss by impairing the functionalities of the computer networks. Denial of Service (DoS) and Distributed DoS (DDoS) attacks are two of the most harmful threats to the network functionality. Mobile Ad Hoc Networks (MANET) are even more vulnerable to such attacks. Ad Hoc On-Demand Distance Vector (AODV) is an outstanding wireless routing protocol. However, AODV has significant security vulnerabilities.

Most current proposed security strategies for AODV or other MANET routing protocols require modifications of the protocols, or of the topology, or even both. Fixing the protocol flaws is obvious and straightforward. But it is impractical and infeasible for an operational commercial MANET. To circumscribe the attack traffic by deploying a

large amount of the edge ingress control nodes or clustering the networks is effective. But it is costly and also requires protocol modification in some circumstances.

The dissertation presents the security solution for AODV and AODV-like networks from a novel perspective. The proposed defense system is based on proxy-based overlay architecture. The proxy guard nodes control the service-related traffic, filter the malicious packets and reinforce the legitimate ones. It assumes a strong restriction on any secure modification on the objective MANET infrastructure. The proposed solution assures a minimum impact on the objective system infrastructure or the network communication interface to make it easy to implement and update, while providing an acceptable secure protection against DDoS attacks, such as Router Requirement (RREQ) flooding, data flooding and black-hole.

ACKNOWLEDGEMENTS

The author would like to thank Dr. John A. Hamilton, Jr., Dr. Martin Carlisle and Dr. Yu Wang for the direction and help on the dissertation and the research. Thanks are also due to my dear parents and my great sister Xin Yu.

Style manual or journal used ACM.

Computer software used Microsoft Word.

TABLE OF CONTENTS

TABLE OF FIGURES

TABLE OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Background

Security is a weak link of network systems. The malicious usage and attacks have caused tremendous loss by impairing the functionalities of the computer networks. Among all network attacks, Denial of Service (DoS) and Distributed DoS (DDoS) attacks are two of the most harmful threats to network functionality.

Mobile Ad Hoc networks are even more vulnerable to these attacks. Existing MANET routing protocols, such as Ad Hoc On-Demand Distance Vector Routing Protocol (AODV), do not provide enough security defense capacity.

Major research efforts have been taken to solve this problem. But most of the proposed solutions are not feasible or practical for the operating MANETs. Because some or all nodes of the MANETs are in a dispersal pattern, or the nodes could be possessed by individuals, it is difficult to apply a network-wide security upgrade. Not only operating MANETs but also any upcoming or planned MANETs face this problem. Though an upcoming MANET can apply the up to date defense strategy, any unpredictable, unforeseen DDoS attack technique in the future can threaten the network and put it in the same situation of those operating unsafe MANETs.

1.2 Thesis Summary

The proposed security strategy limits DDoS attacks on the operating AODV-based MANETs. The security architecture is composed of one service provider and multiple proxies. The service provider announces an artificial IP address for the service. The proxy nodes update the route for all the other system nodes. These proxies listen and reply with the RREQ addresses toward the artificial IP. And the proxy tunnels all the following service traffic to the service provider. The service provider periodically changes its IP address, and the proxies update the route to the service provider. The mechanism prevents the malicious scanning, eavesdropping, and penetration attacks. When a DDoS attack takes place, the congested proxy will break the link to the service provider actively or reactively. And because the legitimate requests are not automatically generated traffic, they will resend the RREQ to the service provider according to the definition of AODV. A further distant but available proxy can receive and respond to these RREQ. The route will be rebuilt and maintained.

1.3 Thesis Contribution

Unlike other recently proposed MANET security strategies, this dissertation presents the solution from a novel attitude. It assumes a strong restriction on any secure modification on the objective MANET infrastructure. The security strategy does not require involvement or support of other nodes; nor does it change the AODV protocol on any customer node. In another words, the communication interface between an architecture node and its neighbors agrees to the definition of AODV protocol. The proposed solution assures a minimum impact on the objective system infrastructure and

topology to make it easy to implement and update, while providing an acceptable secure protection against DDoS attacks.

1.4 Dissertation Structure

The remainder of the dissertation is structured as follows. Chapter 2 provides a literature review of the background and current research progress. Chapter 3 provides the details of the proposed strategy. Chapter 4 describes the experiment and simulation design. Chapter 5 displays the simulation and experiment results. Chapter 6 summarizes the research and discusses the future work.

CHAPTER 2

LITERATURE REVIEW

2.1 Network Security Background

In the history of computer networks, the crucial security properties, confidentiality, integrity, and availability, have never been thoroughly ensured. Malicious attacks have caused tremendous loss by impairing the functionalities of the computer networks since the beginning stage of its development. In 1988, the first network security incident, the "Morris worm", struck the Defense Advanced Research Projects Administration (DARPANET) and brought down 10 percent of the 60,000 node network [1]. From then on, network attacks have kept up with the evolution of computer networks, and each new type of attack brings only broad and more severe damage. According to the Computer Emergency Response Team (CERT) [2] reports, the number of incidents in 1988 was only 6, while in 2003 it was 137,529. And because the "attacks against Internet-connected systems have become so commonplace, as of 2004, CERT will no longer publish the number of incidents reported" [3]. The essential motives of the network attack change over time too. In the early years, the attackers were more interested in discovering the networks' constitution and to display their personal hacking skills. Now the intrusion motives come more from financial, political, and military objectives [4].

When TCP/IP protocols and the Internet were invented, the main goal was to build a stable and robust data communication linkage among Department of Defense and several universities. The users were assumed to be a trustable restricted group. Security was not the most important design issue. The Internet has been growing at an exponential speed in less than two decades. The earlier less-prior system limitation becomes the fatal target of malicious attacks nowadays. Also because of the growth in TCP/IP networks, the required deployment, scale, performance concerns and requires backward compatibility restrict any practical security renovation or even some obvious improvement at the network layer of the Internet [5]. Any proposed security strategy has to consider and be compatible with other possibly unprotected peers of the networks.

Unfortunately, it is déjà vu for the development of the wireless networks. Wireless networks have more sound reasons not to put the security issue at the top of all research goals. Compared to wired networks counterpoint, wireless networks are more fragile, while having much less resources to put in for protection and defense [6]. But, fortunately, for the same reason of the hardware restriction and premature technology, also because of the highly independent system implementation, individuals do not easily deploy the attacks into a specific wireless network. Meanwhile, the security concern has been more and more taken into account in wireless networks research and development. Plenty of research into defensive strategies have already been proposed before any targeted attacks could actually take action [7, 8].

2.2 Denial-of-Service (DoS) Attacks in Wired Networks and the Internet

Among all the Internet attacks, DoS attacks are one of the most significant threats to network functionality. DoS attacks exhaust the network's resource of a specific Internet service or system so that the legitimate users lose the access to the resource [9]. The first DoS attack case happened on Panix, the ISP (Internet Service Provider) of New York City area on September 6, 1996 [10]. According to the 2004 FBI Report on Cybercrime, the total reported costs of DoS attacks were over $26 million. Denial of service was the top source of financial loss due to cybercrime in 2004 [11]. DoS attacks exploit the vulnerabilities of the network protocol architecture. They do not need complicated technology, and they are very easy for attackers to launch, but very hard for victims to prevent and track back.

According to the attack trail, DoS attacks are classified as direct and reflected types (Figure 1) [12].

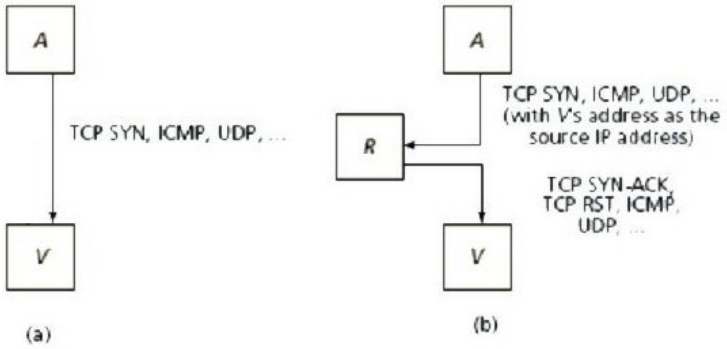


Figure 1. Direct and Reflected Attacks

Some specific DoS types are list below.

- Apache2. The attacker sends a service request with many HTTP headers to a victim Apache Web server. After many such requests, the victim slows down and eventually crashes [13].
- ARP Poison. The attacker has access to the victim LAN. It responds to "arp-who-has" requests in the network as soon as possible and provides the victim with wrong MAC addresses to mislead them [14].
- Back. The attacker floods an Apache Web server with the requests containing a large number of frontslash characters in the URL. The victim server cannot process other legitimate requests, as the server tries to process these attacking requests [13].
- CrashIIS. The attacker sends a malformed GET request to the victim MS WinNT IIS Web server to cause the server to crash [13].
- DoSNuke. The attacker floods the victim MS WinNT with "out-of-band" packets. As a result, the victim is crashed, and turns into a "blue screen" [13].
- Land. The attacker sends the victim a spoofed TCP SYN packet in which the source and destination addresses are same. It may lock some specific types of systems [15].
- Mailbomb. The attacker sends a large amount of messages to overflow and fail the victim's mail queue and system [16].
- SYN Flooding. The attack uses the weakness of the TCP handshake. It sends an abundance of TCP SYN packets to the victim. The victim opens a lot of TCP connections and responds with ACK. But the attacker does not finish the handshake, which, in result, causes the half-open TCP connections to

overflow the victim's incoming queue. SYN Flooding does not target specific Operating System, so it may attack any system supporting TCP protocol (Figure 2) [17].

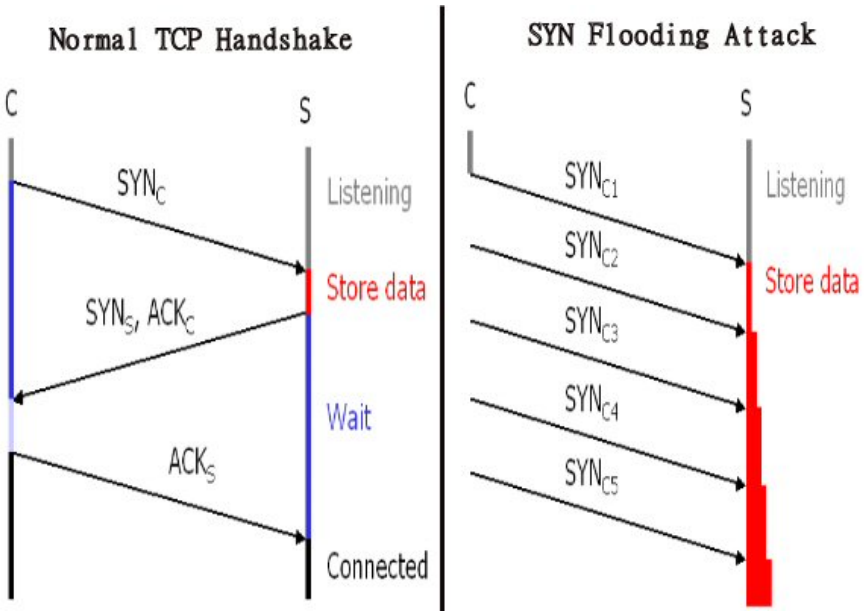


Figure 2. SYN Flooding Attack

- Ping of Death. The attacker sends the victim oversized IP packets, which contain more than 65,536 bytes. It may cause the victim machine to crash [18].
- Process Table. The attacker sends an abundance of uncompleted connections to the victim server. The victim will create a new process for each connection until it cannot serve any more requests.
- Smurf Attack. The attacker sends the broadcast address an abundance of Internet Control Message Protocol (ICMP) "echo-request" packets, which has the victim's IP as the source address. The victim will be flooded with ICMP "echo-reply" packets [19].

- SSH Process Table. The attacker overflows the SSH daemon in the victim system. It is similar to the process table attacks.
- TCP Reset. The attacker listens the traffic for the "tcpconnection" requests to the victim. Once such a request is found, the attacker sends a spoofed TCP RESET packet to the victim and obliges it to stop the TCP connection [20].
- Teardrop. The attacker creates a stream of IP fragments with their offset field overlapped. The victim may crash when trying to reassemble these malformed fragments [15].
- UDP Packet Storm. The attacker spoofs a start packet and builds a connection between two victim nodes, which provide a type of UDP output services (such as "chargen" or “echo”) to generate numerous traffic into the network [21].

2.3 Distributed DoS (DDoS) Attacks in Wired Networks and the Internet

DDoS attacks first appeared in the summer of 1999. The victims were several high capacity commercial and educational websites [22]. The characteristics of Distributed Denial-of-Service (DDoS) are “WMD” (Wide, Massive, Dissemination). DDoS attacks are more powerful, leading to greater damage and easier to perform by Trojan horses, but harder to be prevented and traced back because of the numerous compromised civilian nodes. DDoS attackers user a group of compromised nodes (zombies) to carry on a “large-scale coordinated” attack against the target nodes, where compromised nodes are called the “secondary victims”, and the target nodes are called the “primary victims”. DDoS traffic stream is not unusually high near the attack sources, so it is hard to detect DDoS attacks in the early stages when the attack traffic is still close

to the source. This characteristic provides a good concealment to the real attacker. DDoS traffic streams congest the victim node and often, the intermediate nodes ahead of the victim. This characteristic provides the maximum damage effect to the victim. The victim could be overwhelmed before it takes any defensive action, or the intermediate nodes ahead of the victim may be crashed and the victim will not receive any warning.

There are many tools now on the Internet making a DDoS attack much easier to launch. These tools are classified as either Agent-Handler model or the IRC-based model [23]. With Agent-Handler tools, such as Trinoo [24], Tribe Flood Network (TFN) [24], mstream [25] and so on, an attacker can command the compromised nodes to generate a flooding attack (Figure 3). Stacheldraht [26] combines the features of both Trinoo and TFN, and it encrypts the communication inside the attack system.

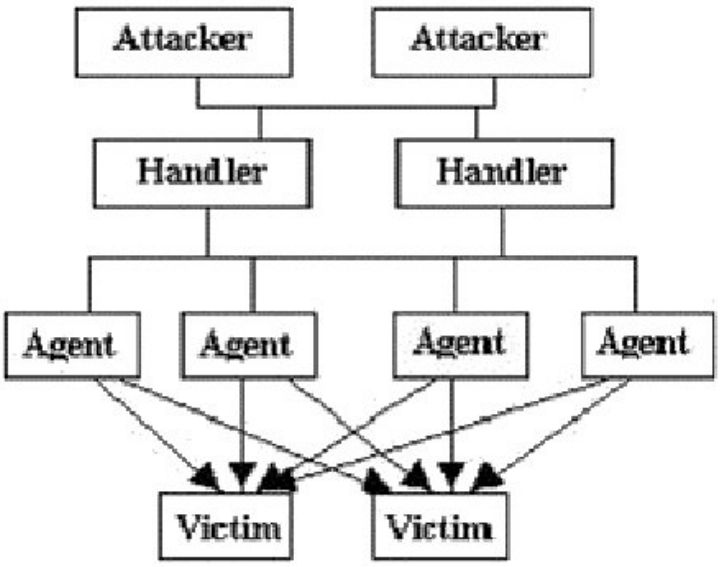


Figure 3. Agent-Handler DDoS attack

IRC-based Botnet type tools have become popular to deploy DDoS attacks [27]. A report from Gartner, Inc. predicts that by 2007, fifty percent of Internet-active companies without attack prevention strategies will suffer financial or service losses by the botnets [27, 28]. The Botnets are often an IRC program, which is installed on the compromised hosts by attackers. Eggdrop [29] and Agobot [30] are two well-known Botnet tools. The Agent-Handler commands have easily detectable patterns, while IRC-based Botnets communication is more flexible and concealed. Except to launch the DDoS attacks, Botnets are also used to install Advertisement Addons to the web browsers, identity theft, spamming, and other malicious activities [27]. To illustrate the jeopardy of the Botnet, the Honeynet project claims that they observed 226,585 unique IP addresses compromised to the Botnet attackers in only few months [27].

2.4 DoS and DDoS Defense in Wired Networks and Internet

There is no one comprehensive defense for all types of DoS attacks [31]. One thorough but simple taxonomy of DoS defense strategies is presented by Bharat Bhargava (Figure 4) [32]. Another taxonomy is presented by Jelena Mirkovic and Peter Reiher [33].

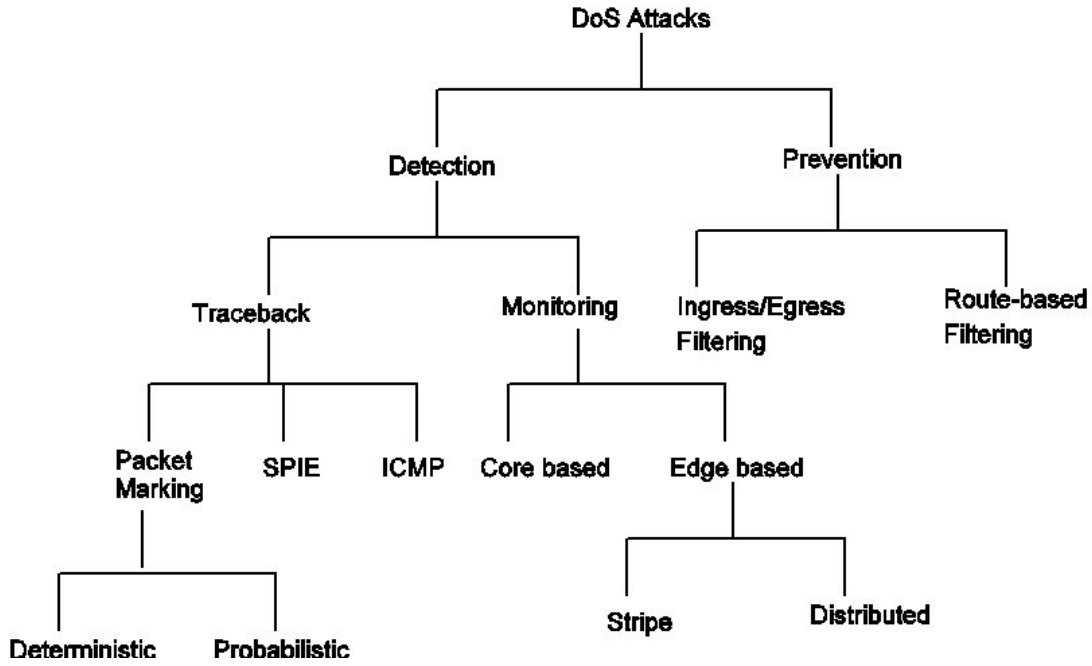


Figure 4. Taxonomy of DoS defense

2.4.1 DoS Attack Detection

To detect an attack in progressing as early as possible is prerequisite to any prevention strategy. This is based on the comprehensive monitoring by the intrusion detection systems.

Different intrusion detection mechanisms are proposed to put on different positions of the network. In "core based" monitoring scheme, the ingress router sends probe packets along the same path as the data packet, then the egress router pick these probe packets and computes the network states [34]. In edge based monitoring scheme, link loss ratio inside a domain is inferred without relying on core routers [35-37].

The further defense step after detected the ongoing DoS attacks is to locate the attackers. In the ICMP traceback scheme, the distributed routers sample the forwarded packets with a very low probability [38]. And these sample packets provide the victim

the reverse paths leading to the attackers. In Source Path Isolation Engine (SPIE) scheme, hash-based system records each single recent IP packet for the topology graph that can reveal the attackers [39]. In packet marking scheme, routers put the hop information inside the packet header to stamp the path [40]. The probabilistic packet marking (PPM) improves the efficiency of routers by marking only a small fraction of packets [41]. Packet marking puts big overhead on routers. The limitation of PPM is that it cannot handle DDoS attacks. And both need to deploy new IP protocol changes on routers in the network.

2.4.2 Prevention

Using a firewall to filter incoming and outgoing network traffic is the classic prevention method [42]. Ingress/Egress filtering belongs to this type [43, 44]. Route-based filtering uses routing information to validate a packet on its source and destination addresses [45]. The routers in these models are required to know the topology of the network. Packet filtering can only accept or deny a packet, so it does not work on all type of DoS attacks, such as intruding purpose packets or attacks using system flaws.

To improve the defense capacity against DoS attacks, sufficient network resources, packet filtering mechanisms, load-balancing servers, and other measures should be set up in advance. A network can counter the attack effort by having sufficient network resources including bandwidth, memory and processor speed. Often, the packet-processing ability is more important than the bandwidth [31]. Distributed systems tolerate attacks better. One plan is to distribute network load and web content to several servers [9]. Another plan is to distribute the load and content to multiple operating

systems (OS) to survive OS-specific attacks [31]. Staying up-to-date with the newest security patches may also prevent OS-specific attacks.

2.5 Mobile Ad Hoc Networks

A Mobile Ad Hoc Network (MANET) is a decentralized, self-organizing, and adaptive gathering of independent mobile nodes, which are communicating over wireless links [46]. Each node is both a network user and a router. Because of the mobility of each node, the network topology may change frequently and be unpredictable. MANETs are attractive in military or civil situations where a rapid deployment and dynamic adaptation are required. Comparing with wired networks, MANETs offer advantages such as mobility, flexibility, and no fixed infrastructure required, but there are more research challenges for MANETs:

- The limited radio signal range requires a wireless node to stay within the network.
- The radio signal could be blocked or absorbed by some objects, and interfered or reflected by some others. The radio signals in the same band from the nearby nodes would collide each other. The range restriction and possible collisions makes packet loss more likely. Therefore, the bandwidth is often lower than that of a wired network. But some new standards (e.g. 802.11 Wi-Fi and 802.16 WiMAX) claim wireless bandwidth comparable to those of Ethernet [47].
- The mobile nodes have limited battery and computation power. Some power-saving strategies may be applied. The nodes may listen to the receivers

periodically; therefore, the nodes may not receive the signals in time. They may also need time to wake up and get ready for the communication. This may lead to high communication latency.

- Because of the mobility and flexibility of the nodes, it is required to quickly adapt to the change of the network topology and look up the specific node. A commercial MANET needs to implement a QoS solution for the traffic.
- Because of the mobility and the dynamic construction of the ad hoc nodes, one essential research topic of MANETs is about accurate and efficient service discovery, lookup and verification methods [48-50].

Some security challenges to MANETs are:

- MANETs use wireless media for transmission, which introduces security flaws to the networks. Basically any one with the proper equipment and knowledge of the current network topology and the protocols may obtain access to the network. Both active and passive attacks such as impersonation [51], eavesdropping [52], message redirection, and traffic analysis, can be performed by an adversary.
- In specific scenarios, MANET nodes may be scattered over a large area. Some nodes or network components may be unmonitored or hard to monitor, and exposed to the physical attacks.
- Because MANETs do not have any central authority, this is a major barrier to security. The security mechanisms employed in wired networks, such as Public Key Management, Node Authentication, and Determination of Node

Behavior, are in fact very difficult to achieve without any central administration.

- Ad hoc networks are highly dynamic in nature. Node joins and departures are not predictable. Moreover, network topology is always changing in Ad Hoc networks. Therefore any static security mechanism will not be applicable in MANETs. In other words, security primitives must be dynamically adjusted to cope with the network. This is a daunting task [53].

2.6 Basic MANET Routing Protocols

MANET Routing protocols can be classified as table-driven/proactive and on-demand/reactive (Figure 5). Proactive protocols maintain up-to-date network-wide routing information in advance. The routing maintenance packets are propagated throughout the network as changes in the topology occur. Proactive protocols trade the periodic routing maintenance overhead for immediate availability. It is an advantage only if there are many route requests within a short period of time. Reactive protocols do not execute a routing update until the communication needs it. When a route is needed, the source node initiates a route discovery process to the destination. Once established, the route must be maintained until it is no longer needed or the destination node becomes inaccessible. Reactive protocols trade the routing update delay for less system overhead, and less power consumption, which is critical to battery life in the MANET environment.

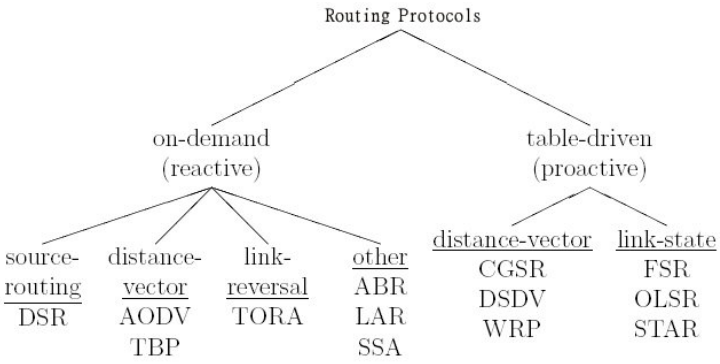


Figure 5. MANET Routing protocols [54]

2.6.1 Proactive/Table-Driven routing protocols

Routes from proactive routing protocols are built up before one is needed. Routing information is kept up-to-date in either event-driven or periodical manner, which requires a significant communication and calculation workload.

- Wireless Routing Protocol (WRP) [55] is a Distance Vector routing protocol. It eliminates the "Count-to-infinity" problem.

  Fisheye State Routing [56] based on link state routing and it maintains a full topology map at each node. Therefore it can immediately provide route information when needed. The fisheye scope technique allows exchanging link state messages at different intervals for nodes within different fisheye scope distance, which helps to reduce the size of the link state message (Figure 6).

  It maintains a flat addressing scheme and topology map, which limits the scalability of the networks. It also introduces high routing table storage complexity and the processing overhead. FSR does not provide any form of security.

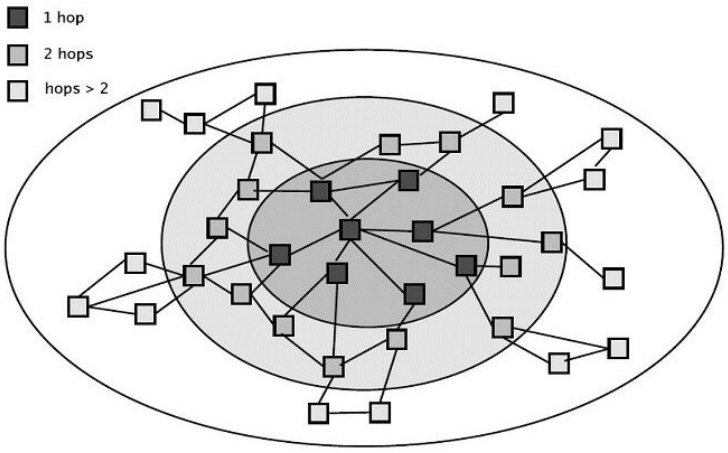
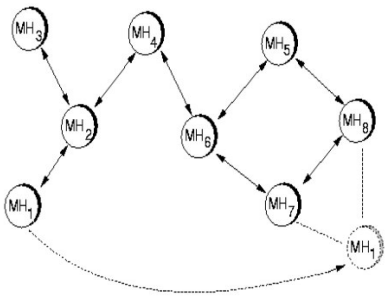


Figure 6. Scope of Fisheye [56]

- Destination Sequenced Distance Vector (DSDV) [57] is a table driven hop-by-hop distance vector routing method. Each node maintains a complete routing table, which contains entries for every other reachable node. Nodes pass their routing tables to neighbors periodically. Routing tables are updated with using the standard distance vector algorithm (Figure 7).

  DSDV responds to routing changes quickly, and guarantees loop-free paths. But it requires a high volume of maintenance traffic to keep the topology updated.

- Optimized Link State Routing Protocol (OLSR) [58] is an optimization of link-state routing protocol. The update packets are forwarded by the relay nodes, which are the direct neighbors. This idea (multi-point relays, MPR) reduces the network traffic but introduces more computation and complexity.



Destination	NextHop	Metric	Sequence number	Install	Flags	Stable_data
MH_1	MH_2	2	S406_MH_1	T001_MH_4		Ptr1_MH_1
MH_2	MH_2	1	S128_MH_2	T001_MH_4		Ptr1_MH_2
MH_3	MH_2	2	S564_MH_3	T001_MH_4		Ptr1_MH_3
MH_4	MH_4	0	S710_MH_4	T001_MH_4		Ptr1_MH_4
MH_5	MH_6	2	S392_MH_5	T002_MH_4		Ptr1_MH_5
MH_6	MH_6	1	S076_MH_6	T001_MH_4		Ptr1_MH_6
MH_7	MH_6	2	S128_MH_7	T002_MH_4		Ptr1_MH_7
MH_8	MH_6	3	S050_MH_8	T002_MH_4		Ptr1_MH_8

Table 1: Structure of the MH_4 forwarding table

Destination	NextHop	Metric	Sequence number	Install	Flags	Stable_data
MH_1	MH_6	3	S516_MH_1	T810_MH_4	M	Ptr1_MH_1
MH_2	MH_2	1	S238_MH_2	T001_MH_4		Ptr1_MH_2
MH_3	MH_2	2	S674_MH_3	T001_MH_4		Ptr1_MH_3
MH_4	MH_4	0	S820_MH_4	T001_MH_4		Ptr1_MH_4
MH_5	MH_6	2	S502_MH_5	T002_MH_4		Ptr1_MH_5
MH_6	MH_6	1	S186_MH_6	T001_MH_4		Ptr1_MH_6
MH_7	MH_6	2	S238_MH_7	T002_MH_4		Ptr1_MH_7
MH_8	MH_6	3	S160_MH_8	T002_MH_4		Ptr1_MH_8

Table 3: MH_4 forwarding table (updated)

Figure 7. DSDV Operation [57]

- Clusterhead Gateway Switch Routing (CGSR) [59] divides the network into clusters, and a clusterhead is elected for each cluster. The clusterheads are in charge of broadcasting within the cluster, forwarding messages and dynamic channel scheduling (Figure 8).

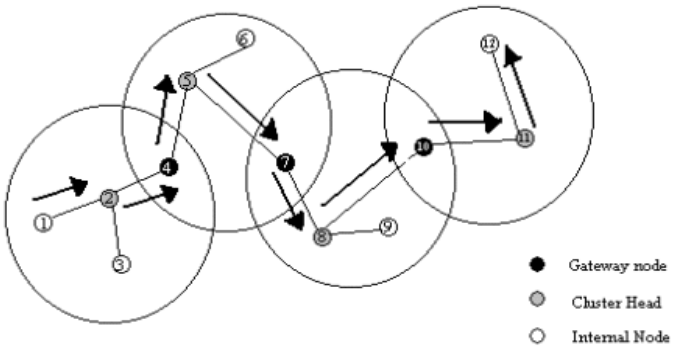


Figure 8. CGSR Operation [59]

CGSR reduces the routing table and makes the routing quicker so that the paths are more stable, which in all makes the protocol more efficient.

2.6.2 Reactive/On-Demand routing protocols

Reactive routing protocols update routes only when they are needed and only to those interested nodes.

- Dynamic Source Routing (DSR) [60] is based on the Link-State-Algorithms. The receiver floods the network with route requests, and the sender determines the whole path and lists it in the packet header (Figure 9).

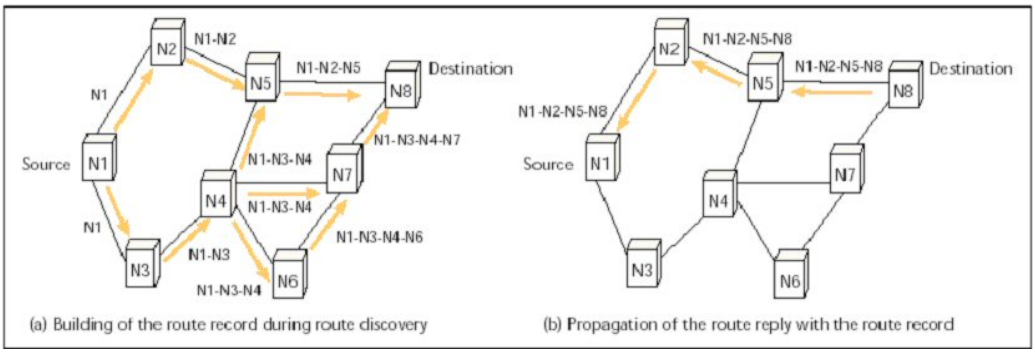


Figure 9. DSR Operation [60]

DSR is restricted on scalability and mobility. The diameter of the network is only 5 hops to 10 hops, and the nodes should move at only a moderate speed.

- Ad Hoc On-Demand Distance Vector Routing Protocol (AODV) [61] is based on DSDV and DSR. A source node floods RREQ (Route Request) messages, and a destination node sends a RREP (Route reply) along the path with the most recent sequence number (Figure 10 and Figure 11). An AODV node only records the next hop of the route. The packets need not store the route as they do in DSR.

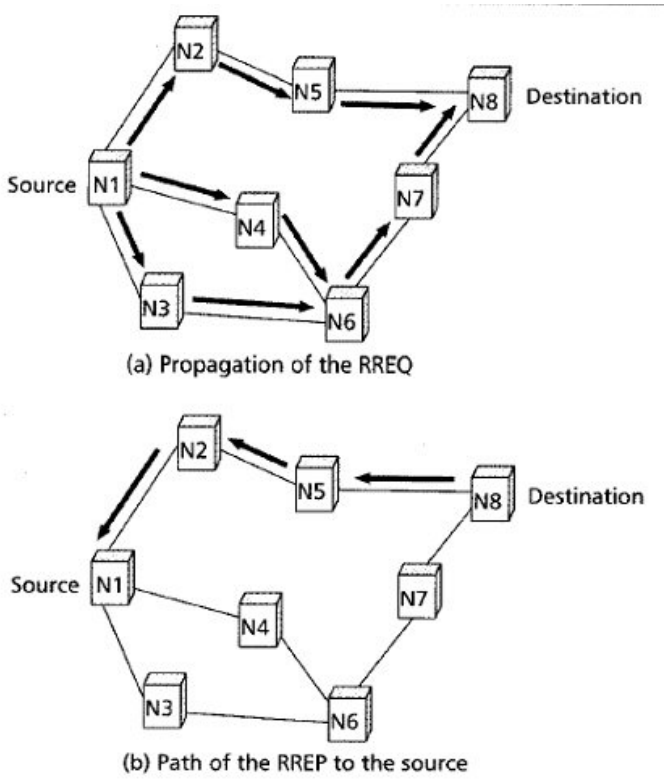


Figure 10. AODV Operation [61]

	Route Request			Route Reply	
1	$S \rightarrow *:$	$\langle \text{RREQ}, S, D, 0, S \rangle$		$D \rightarrow C:$	$\langle \underline{\text{RREP}}, S, D, \underline{0}, \underline{D} \rangle$
2	$A:$	$d[S] = 1, n[S] = S$	6	$C:$	$d[D] = 1, n[D] = D$
	$A \rightarrow *:$	$\langle \text{RREQ}, S, D, \underline{1}, \underline{A} \rangle$		$C \rightarrow B:$	$\langle \text{RREP}, S, D, \underline{1}, \underline{C} \rangle$
3	$B:$	$d[S] = 2, n[S] = A$	7	$B:$	$d[D] = 2, n[D] = C$
	$B \rightarrow *:$	$\langle \text{RREQ}, S, D, \underline{2}, \underline{B} \rangle$		$B \rightarrow A:$	$\langle \text{RREP}, S, D, \underline{2}, \underline{B} \rangle$
4	$C:$	$d[S] = 3, n[S] = B$	8	$A:$	$d[D] = 3, n[D] = B$
	$C \rightarrow *:$	$\langle \text{RREQ}, S, D, \underline{3}, \underline{C} \rangle$		$A \rightarrow S:$	$\langle \text{RREP}, S, D, \underline{3}, \underline{A} \rangle$
5	$D:$	$d[S] = 4, n[S] = C$	9	$S:$	$d[D] = 4, n[D] = A$

Figure 11. AODV RREQ and RREP Example [61]

AODV has a scalability problem because the size of the routing table grows linearly with the number of the nodes [62]. The movement of the nodes may

trigger frequent flood-searches, which is the combination of the overhead of DSDV state maintenance plus DSR flooding. AODV does not have any security mechanisms, so it is vulnerable to many attacks [63]. AODV is discussed further in 2.7.5 AODV Attacks.

AODV has better performance than other MANET routing protocols [64]. It is also the most discussed, compared, and extended protocol. Some research projects focus on AODV extension and improvement. Multicast AODV (MAODV) [65] is a multicast group based AODV, which can perform unicasting, multicasting and broadcasting. Power-aware AODV [66] focuses on extending the battery life in the AODV environment. Multipath AODV [67] uses a pair of link-disjoint paths to improve the fault tolerance of the route. Mobile agents based AODV [68] uses ant-like technology to save the network resource. Secure AODV (SAODV) uses the public key algorithm and signature method to validate the traffic.

- Temporarily Ordered Routing Algorithm (TORA) [69] provides multiple routes from a source node to a destination node. Each intermediate node along the route has "height" according to the distance to the destination. Closer node has less height. By this way, a directed acyclic graph is constructed, which is described as water flowing downhill (Figure 12). TORA uses a "single pass" strategy, by which all route maintenance tasks can be combined into one event [70]. TORA has scalability problems, and is not able to adapt to fast changes in the network without significant overhead. The packet throughput of TORA is low [71].

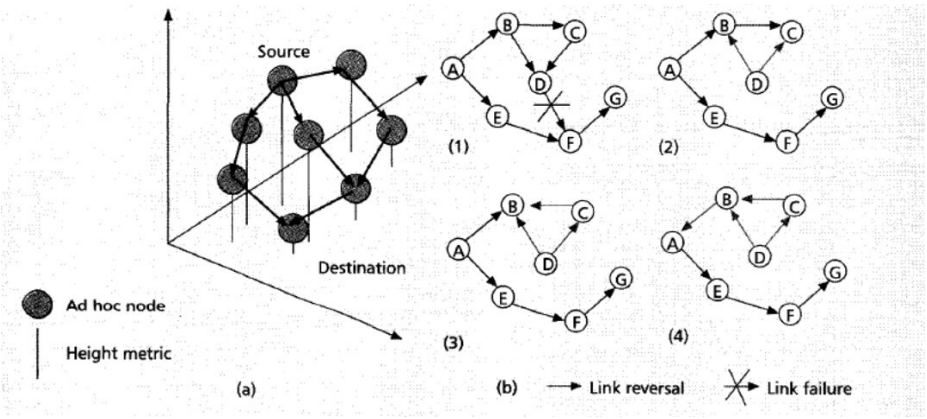


Figure 12. TORA Operation [69]

- There are some other on-demand routing protocols. CEDAR [72] is a hierarchical routing method for quickly and effectively reacting to the dynamics of the network. Signal Stability-Based Adaptive Routing (SSA) [73] takes the signal strength in to account to find the link with strongest signal instead of the one with shortest path.

2.6.3 Hybrid of Proactive/Reactive

Zone Routing Protocol (ZRP) [74] proposes a region ("zone") on each node. A node needs only the knowledge about the routing inside the zone, which requires a smaller routing table; and has a routing lookup only between the node and its perimeter nodes, which is faster and takes less resources.

Sharp Hybrid Adaptive Routing Protocol (SHARP) [75] trades off between proactive and reactive methods. It also uses the "zone" principle. It applies proactive routing to the neighborhood inside the zone, and applies reactive routing to the remote destination outside the zone.

2.6.4 Other routing protocol categories

Geographic routing protocols [76] suggests GPS support to the nodes, so that they need only directional routing lookup, or furthermore, know the accurate geographic position of the destination ahead of the communication. The goal is to avoid the routing broadcasting and delay.

- Location-Aided Routing Protocol (LAR) [77] is an on-demand protocol. It reduces the routing overhead by imposing the node location information.
- Greedy Perimeter Stateless Routing (GPSR) [78] combines Greedy Packet Forwarding and Perimeter Forwarding based on the geographic position of the destination (Figure 13).

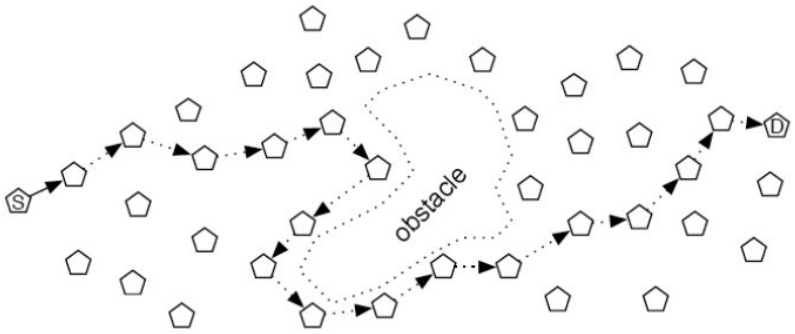


Figure 13. GPSR Routing Example [78]

Clustering methods [79], including CEDAR and CGSR mentioned above, try to solve the scalability problem of MANETs by managing the hierarchical information instead of the information of all the nodes.

LANMAR [80] and “Hierarchical Approach to Position-Based” [81] are two protocols that combine both the geographic and the clustering methods.

2.7 DoS and DDoS Attacks on MANETs

Attacks on MANETs come in many varieties and they can be classified based on different aspects.

2.7.1 Legitimate Based Classification

According to the legitimate status of a node, an attack could be external or internal. The external attacks are committed by nodes that are not legal members of the network, while the internal attacks are from a compromised member inside the network. The internal attacks are not easy to prevent or detect. These attackers are aware of the security strategies, and are even protected by them. The internal attacks pose a higher threat to the network.

2.7.2 Interaction Based Classification

In terms of interaction, an attack could be passive or active. Passive attacks do not disrupt the communication. Instead, they intercept and capture the packets to read the information. On the other hand, active attackers inject packets into the network to interfere or interrupt the network communication, overload the network traffic; fake the legitimate node or package, obstruct the operation or cut off certain nodes from their neighbors so they can not use the network services effectively anymore. DoS or DDoS are active attacks (Figure 14).

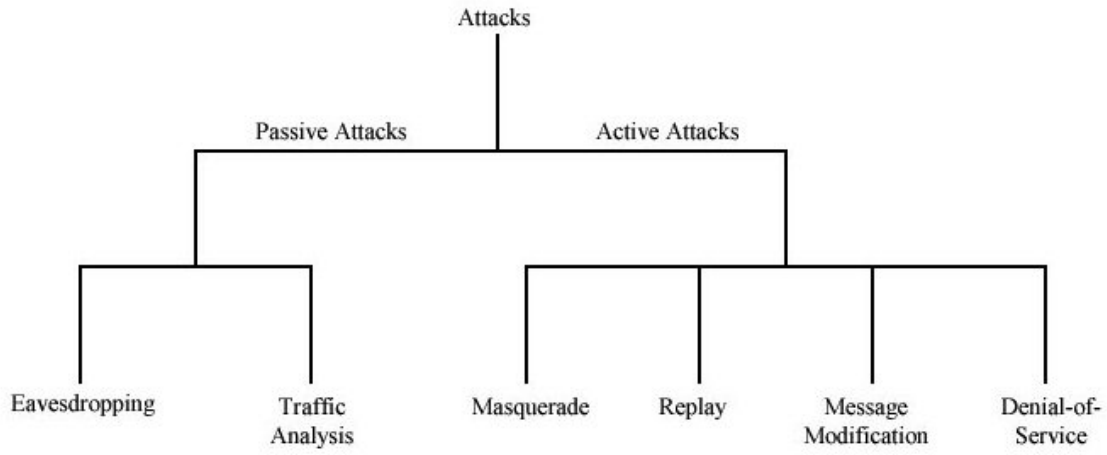


Figure 14. Taxonomy of MANET Attacks

2.7.3 Network Protocol Stack Based Attack Classification

Attacks could also be classified according to the target layer in the protocol stack (Figure 15) [82].

Stack Layer	Attacks
Application	Backdoor, Virus, Data corruption or deletion, Repudiation
Transport	Desynchronization, Session hijacking, SYN flooding
Network	Blackhole, Byzantine, Flooding, Location disclosure, Misdirection, packet dropping, Resource consumption (Sleep deprivation), Rushing, Selfish, Spoofing, Wormhole
Link	Collision, Disruption MAC (802.11), Exhausting, Monitoring (Traffic analysis), Unfairness, WEP weakness
Physical	Eavesdropping, Interceptions, Jamming, Tampering
Multi-layer attacks	DoS, impersonation, replay, man-in-the-middle

Figure 15. MANET Protocol Stack and DoS Attacks [82]

2.7.3.1 Physical Layer Attacks

By targeting the physical layer of a wireless network or a wireless node, an attacker can easily intercept and read the message contents from open radio signals [83,

84]. An attacker can jam or interfere the communication by generating powerful transmissions to overwhelm the target signals. The jamming signals do not follow the protocol definition, and they can be meaningless random noise and pulse [85].

2.7.3.2 Link Layer Attacks

By targeting the link layer, an attacker can generate meaningless random packets to grab the channel and cause collisions [86]. In this situation, if the impacted node keeps trying to resend the packet, it will exhaust its power supply; The attacker can passively eavesdrop on the link layer packets; The link layer security protocol WEP is vulnerable too, the initialization vector (IV) flaw in the WEP protocol makes it easier for an attacker to launch a cryptanalytic type attack [87].

2.7.3.3 Network Layer Attacks

Coming along with many new routing protocols introduced to the MANETs, many new types of attacks were presented to target these specific protocols.

- "Black hole" attacks Distance-Vector type routing protocols [88]. A black hole attacker responds to all RREQ with a shortest route RREP. After the attacker grabs the route, it may drop all the packets, or selectively forward some of the packets to hide the malicious nature. It is also the first step in the man-in-the-middle attacks (Figure 16) [89].

  Cooperative black hole attacks over AODV and defense are discussed in [90].

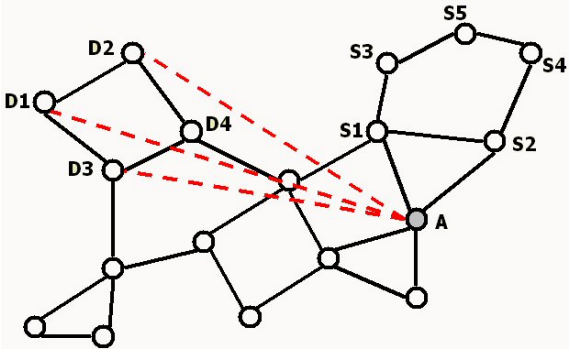


Figure 16. Black hole attack, Attacker A claims to have shortest route to D1, D2, and D3

- "Byzantine" attackers respond to the RREQ with wrong route information to disrupt or degrade the routing services, such as creating routing loops, forwarding packets through non-optimal paths, or selectively dropping packets [91].
- Flooding methods used by DoS and DDoS attackers in wired networks have the same effect on the MANET environment [92].
- "Location disclosure" attackers disclose the security-sensitive location information of nodes or the topology of the network [63].
- "Misdirection" attackers lead the packets to a wrong way and toward the victim. Similar to Smurf attacks [19].
- "Packet dropping" attackers disrupt the network communication, and they are very hard to detect. This type of attack is often working along with other attack methods to amplify the damage [93].
- "Resource consumption" or so-called "Sleep deprivation" attackers try to waste the power of the legitimate nodes by requesting excessive route

discovery, forwarding useless packets to the victim node, or endlessly "dangling" useless packets between two distant attackers.

- "Rushing" attackers have more power and quicker links than legitimate nodes. They may forward the RREQ and RREP faster. By this way, they are always involved in the routes (Figure 17) [94].

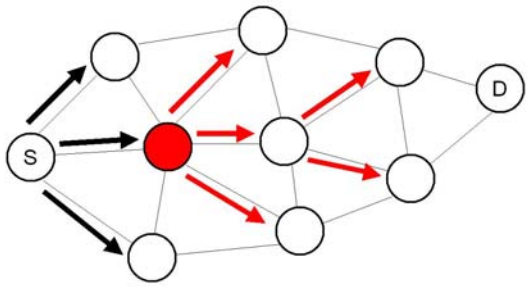


Figure 17. Rushing Attack

- "Selfish" nodes use the network but do not cooperate. They save the battery life, CPU cycles, and other resources for their own packets. Though they do not intend to directly damage other nodes, the result is less damaging inefficient networking [95].
- "Spoofing" attackers impersonate a legitimate node to misrepresent the network topology to cause network loops or partitions [96].
- "Wormhole" attackers forward packets between each other by a tunnel instead of hop based routing method as defined by the protocol [97]. Routing may be disrupted by tunneled routing control messages. Wormhole attacks are severe threats to MANET on-demand routing protocols. The attack could prevent the discovery of any route other than through the wormhole (Figure 18).

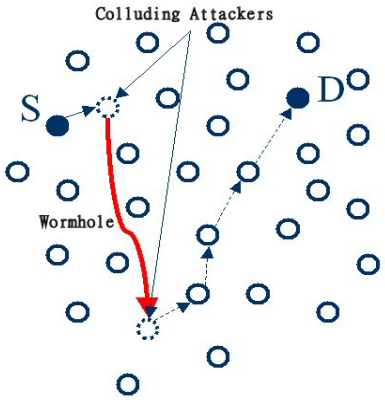


Figure 18. Wormhole Attack

Wormhole attack defense strategies are often based on space or time relativity, such as geographical leashes [97], temporal leashes [97], or a graph theoretic approach [98].

2.7.3.4 Transport Layer Attacks

By targeting the transport layer, a “desynchronization” attacker can break an existing connection between two nodes by sending fabricated packets exceeding the sequence number to either node of the connection. It may result in letting the node keep sending retransmission requests for the missed frames [99]. A “Session Hijacking” attacker impersonates the victim node and takes over the TCP session between the victim and the server [100].

2.7.3.5 Application Layer Attacks

By targeting on the application layer, a “Repudiation” attack is a threat to a business that relies on electronic traffic. Some examples are described in [101-103].

Other application layer attacks, such as viruses, worms, trojans, spywares, backdoor, and data corruption or deletion, target either application layer protocols, such as FTP, HTTP, and SMTP, or applications and data files on the victims [104].

2.7.4 Cryptography Attacks

Some attacks target security leaks on the cryptography primitive of the protocols.

- Digital signature attacks target RSA public-key encryption algorithms [105]. Attackers forge the message signature based on the signature of a legitimate message. Digital signature attacks have three types, known-message, chosen-message, and key-only attacks. The "Known-message" attacker knows a list of messages previously signed by the victim. The "Chosen-message" attacker can choose a specific message that it wants the victim to sign. The "Key-only" attacker knows the public verification algorithm only [106].
- Hash collision attacks target hash algorithms, such as SHA-1, MD4, MD5, HAVAL-128, and RIPEMD, to construct a valid certificate corresponding to the hash collision [107].
- Pseudorandom number attacks reverse engineer the pseudorandom number generators used by the public key mechanisms to break the cryptography [108].

2.7.5 AODV Attacks

Although AODV, as a routing protocol has many advantages, AODV is inherently vulnerable to many attacks (Figure 19) [63].

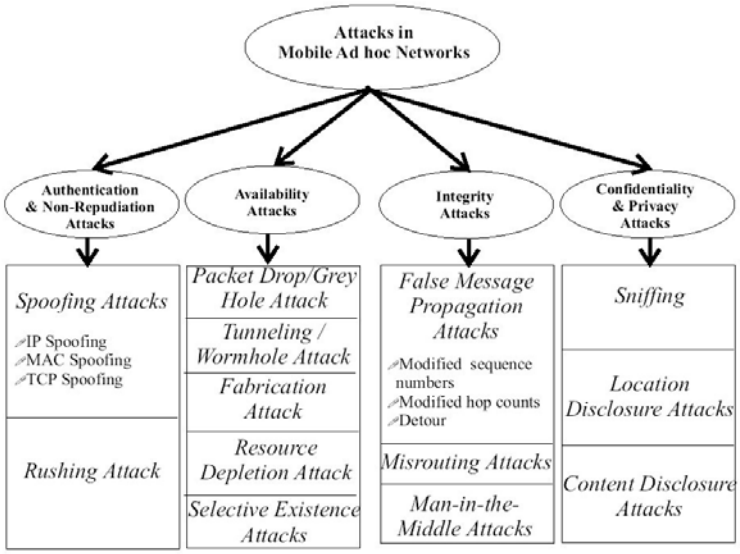


Figure 19. Attacks on AODV in MANETs [63]

2.8 DoS and DDoS Defense in MANETs

2.8.1 Security Aspects of MANETs

MANETs require the four standard security attributes [52].

- Availability, which requires that the system stays up and in a working state, and provides the right access and functionality to each user. This security aspect is the target of DoS or DDoS attacks.
- Confidentiality, which requires that the information will not be read or copied by unauthorized parties. Authentication and other access control techniques are used to achieve this goal.
- Authenticity, which requires that the communication peer is really the legitimate node and is exactly whom we expect to talk to, and that the content of a message is valid.

- Integrity, which requires that communication data between nodes must not be modified by any unauthorized, unanticipated or unintentional parties.

2.8.2 Secure MANET Strategies Classification

A practically operating MANET must consider the tradeoff between the deployment feasibility of a security patch and the system efficiency. And often, the feasibility is considered over the efficiency [109, 110]. The feasibility of a deployment (accessibility and cost) mostly depends on the deployment location. Based on this concept, the defense strategies are classified as attacker-side strategies, victim-side strategies, and intermediate strategies in [111]. This taxonomy makes more practical sense to evaluate a defense strategy than other taxonomies, e.g. activity level or cooperation degree [33]. My dissertation will discuss the proposed solution based on this taxonomy too.

- Attacker-side strategies [43, 44, 112-114]. It puts the ingress control to the edge routers. So that the packets going out into the network are only the legitimate ones. The disadvantage is that it requires not only a large-scale deployment of ingress control, but also the cooperation among the network clusters.
- Victim-side strategies. An authentication system is built up by the victim, then it may let only the legitimate traffic have the access [115, 116], or allocate resources to the requests only after they are authenticated [117]. The disadvantages are that it requires the client to take extra legitimate application

for the access, and DoS congestion may occur before the traffic reaches the victim so the strategy fails.

- Intermediate strategies. It requires multiple intermediate nodes to support the secure system for the target. These intermediate nodes can work as a proxy to forward and filter the packets, or as the traffic monitors to detect the attack patterns. Another usage of the intermediate nodes is to form a multi-tier architecture, which can provide a unified security service (or other MANET services) interface towards client nodes [118].

2.8.3 DDoS defense strategy Examples

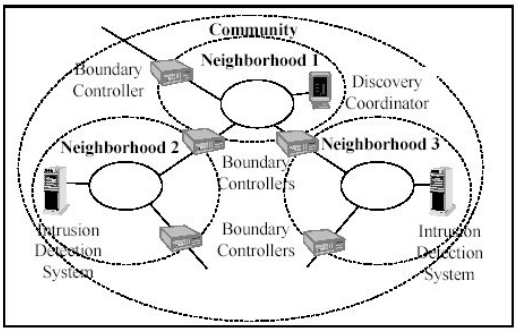
2.8.3.1 Statistical-based Detection and Backtracing

Statistical methods in either packet sampling [119] or packet header marking [39, 120] reduces the detection and backtracing overhead. Packet sampling picks only a small percentage (e.g. 3.3%) of the packets. The processing and storage overhead can be very low. When an attack happens, the flood of the attack packets can rapidly provide enough information for the tracing purpose.

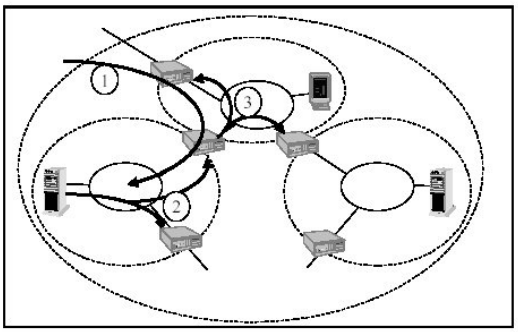
There are other header mark or path mark methods, as described in [121-123]

2.8.3.2 Clustering Networks

Intruder Detection and Isolation Protocol (IDIP) [124] clusters nodes into communities and puts the boundary controllers at the edge of each community. With the help of these boundary controllers, the Intrusion Detection System (IDS) communicates among the clusters to trace back the attacks (Figure 20).



IDIP communities



IDIP intrusion response

Figure 20. Intruder Detection and Isolation Protocol [124]

2.8.3.3 Agent-based Detection

A mobile-agent-based architecture is fully distributed, and is able to randomly select the migration path [92, 125, 126]. It requires a mobile agent platform to be deployed on each node of the system.

2.8.3.4 Authorization

Authorization is a type of "Attacker-side strategy". Strictly authorization filters out all unauthorized traffic [43, 44, 112]. The hierarchical authorization has certain flexibility. The basic idea is that the service provider assigns an authorization key [127] or capability token [128] to the important service requesters, but does nothing to the

regular requesters. When the system is under the attack flood, only the requests with the authorization key can pass and get the service, and all others will be dropped along with the attack traffic. Stateless Internet Flow Filter (SIFF) [127] is an example. This type of strategy works effectively on some particular scenarios, where critical service availability is required. Obviously, it is not for generic purposes.

The Client Puzzles [129], the new Client Puzzles [130] and SYN cookies [131] ask the clients to finish a puzzle before building a connection and allocating the resource.

2.8.3.5 Overlay Architecture

Overlay network is a type of "Intermediate strategy". It is an application-layer virtual architecture over the network infrastructure. Therefore it may be the supplemental security architecture over the existing vulnerable routing protocols. The advantages of the overlay include routing-protocol-independent multi-path support, ingress authorization and enhanced anonymity [132]. There are non-announced overlay strategy over AODV as yet. Some well-known overlay-based strategies include Internet Indirection Infrastructure (i3) [97], Centertrack [133] and Secure Overlay Service (SOS) [134].

- Instead of point-to-pint communication abstraction, i3 provides a rendezvous based abstraction. Each packet is a pair (id, data) where id is an identifier, and data is the packet payload. A receiver R inserts a trigger pair (id, addr) into the overlay network to show the interest of the packet with identifier id (Figure 21). It uses two types of triggers, public triggers and private triggers. The public triggers are used for initialing the rendezvous, and the private

triggers are used for the secure and efficient routing. i3 provides good anonymity and promises the defense on eavesdropping, trigger hijacking, and DoS attacks.

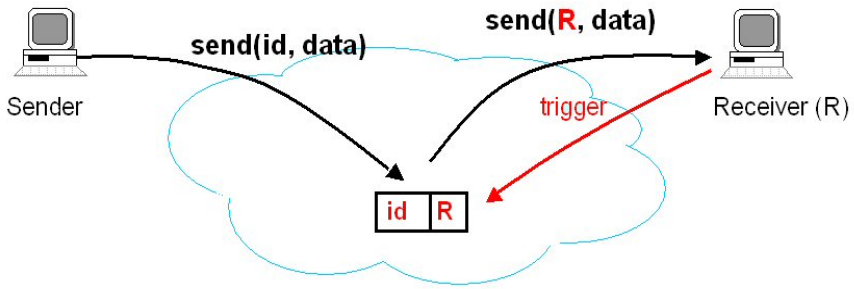


Figure 21. i3 Communication [97]

The initial handshake and network maintenance cause remarkable overhead. This is the performance drawback of i3. Servers must announce their public triggers; otherwise, clients must know them. This reveals the overlay to the DoS attacks. i3 is not implemented in the application layer, so it requires extra modification in the network layer of the server nodes, the client nodes, and all other overlay forwarding nodes.

- Centertrack proposes a traceback method within an overlay network, which consists of edge routers and tracking routers. The edge routers reroute the suspicious traffic to the tracking routers. The tracking routers can distinguish the attack packets, and trace back to the ingress edge router. Centertrack needs high bandwidth for tunneling the rerouted traffic, and the networking

overhead amplifies the effects of a DoS attack. The most significant drawback is that the system cannot effectively trace back the DDoS attacks.

- SOS proactively deploys an overlay in the network, opens a set of access points to the outside, and hides the service provider and other internal overlay nodes. SOS assumes that the service requests are from the nodes knowing the architecture and the access points of the SOS. The secure overlay access point (SOAP) nodes receive the outside requests and tunnel the requests to the forwarding proxy nodes (secret servlets). Secret servlet nodes deliver the requests to the real service provider. The replying routes are vice versa. When the network is under attack, only the legitimate traffic can get into the overlay (Figure 22). Mayday is a generalized version of SOS [135].

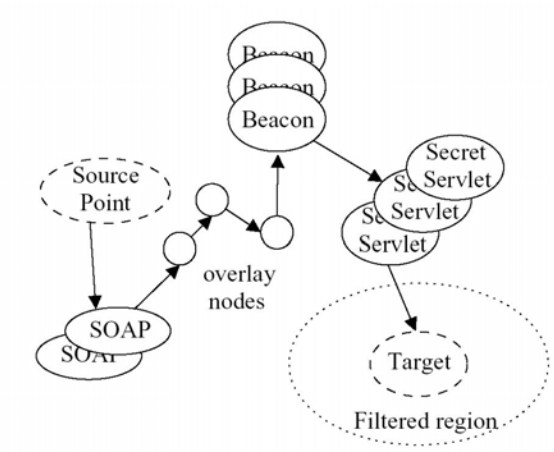


Figure 22. SOS Architecture [134]

SOS has the advantages of the overlay architecture, so it can keep providing the connectivity under the DoS attack [136]. But it has several disadvantages.

1. The attack traffic outside of the overlay is not suppressed; SOS only protects the specific target.
2. Once the attackers trace out the SOAP, they could use spoofing SOAP packets to attack/congest Secret servlets.
3. SOS has no mechanism to trim or balance the traffic, so the system may still crash with more sophisticated break-in traffic.
4. Because SOS service assumes that the both ends of the communication know each other, SOS is not good for the public network services such as google [134].
5. It is proposed for the Internet, so it will have performance, mobility, and deployment problems in MANETs.

2.8.3.6 Anonymity and Privacy Enhancement

- ANonymous On Demand Routing with Untraceable Routes for Mobile Ad-hoc Networks (ANODR) [137] and Anonymity for Users of Ubiquitous Computing [138] focus on another two important security aspects of MANETs, which are route anonymity and location privacy.
  AODV also implies the route anonymity, but multiple collaborative eavesdroppers can figure out the specific route in AODV. ANODR explicitly encrypts the route hop-by-hop, which can totally conceal the routes.
- IP Hopping [139] presents an agile method to avoid attackers' tracking and attack traffic by changing the IP address of the service provider. It may be applied periodically, or when an attack takes place. Because the routes for

both attackers and legitimate users depend on the server's IP, a good IP hopping strategy needs work with the Domain Name Service (DNS) or other IP-Name service. And such a system needs a protection on the DNS server too.

2.9 Practicality Issues of Current Security Solutions

Most of the security strategies mentioned above require modifications to either the network protocols, or to the topology, or even both.

To fix the flaws of the network protocols is an obvious and straightforward solution, which is also the ultimate answer to the DDoS Attack problem. But it is impractical and usually infeasible for an operating commercial MANET. It requires a network-wide node-by-node upgrade. Some nodes may be hard to reach; some unfriendly individual nodes may refuse the upgrade. These troubles may fail the new security functionality of the whole network. Some systems use mobile agent software to spread the upgrade patch through the network and execute the upgrade on each node without physical access to the nodes. But they require a complicated and well-designed agent support system and a large amount of system overhead, and they may introduce new security flaws to the networks. Any network protocol modification may fail the applications. Often an upgrade on the network layer protocols will change the behavior of the link layer, the network layer, and the layers above, and the interfaces between these layers. All applications depending on those old infrastructure behaviors need to adjust to the new changes. The applications' implementation, testing and deployment costs need be considered as well as the overhead of the network protocol upgrades.

To circumscribe the attack traffic at the edge of the networks by deploying a large amount of ingress control nodes or clustering the networks is effective. But it is costly and also requires protocol modification in some circumstances. Service requesters may have to install specific authorization applications to obtain access from the ingress control nodes.

Even a state of the art protocol or a successfully updated protocol may be vulnerable to DDoS attacks utilizing new techniques and have the same troubles like those systems mentioned above do.

SOS is good defense architecture against DDoS in a wired Internet environment, but it is not proper for MANETs because SOS is complicated and not adaptable to mobility.

Therefore, this dissertation attempts to provide an original solution, which assures a minimum impact on network infrastructure and network topology to make it easy and inexpensive to implement and update standalone or overlay-based security strategies, while providing an acceptable secure protection against DDoS attacks.

2.10 Test bed and simulation environment

2.10.1 Introduction of Simulators

The traditional experiment and analysis approaches for a novel network proposal are numerical and analytical methods, computer simulation, and physical measurement [140]. Among the available approaches for wireless network experiments, computer simulation is the most feasible, accurate and realistic. Most protocols or defense

strategies described in this chapter were evaluated in a network simulator, and discussed based on the simulation results.

There are many network simulators. OPNET and ns-2 are two mature and well-known ones. OPNET is a commercial software product, which has full technical support and powerful simulation capacity [141]. The simulation examples can be found in research [111, 136].

Ns-2 is thought to be the most widely used network simulator [142, 143]. Ns-2 is free and powered by abundant up-to-date extensions. The simulation example can be found in research [144]. The data type of time interval in ns-2 is double, which makes event time intervals much less than one millisecond. It provides enough accuracy for my experiments.

Performance comparison shows little difference between ns-2 and OPNET [145]. The dissertation will use ns-2 as the simulator for the reasons mentioned above.

2.10.2 Simulation Levels

The most detailed and accurate simulation is at the packet level. There is little abstraction or concision at this level of simulation, and the detail of each packet over each hop is emulated and logged. Therefore packet level simulation is not efficient for a large-scale network simulation, but it provides the closest result to the real-world experiment.

Fluid-based simulation focuses on more abstract network packet traffic instead of each single packet [146]. When it improves the performance in large-scale network simulation, it is hard to make an accurate emulation of the detailed behavior of the

network traffic. Some research attempts to improve the accuracy to the current fluid-based methods [147-149].

The session level simulator roughly estimates and records the packet delivery time from the sink to receiver. It is used to abstract network scaling issues and multiprotocol composability.

Comparing to session-level or fluid-based counterparts, packet-level simulation provides the most accurate result [150]. The workload and system requirement of a packet-level simulation of a small-scale network environment is also affordable. Therefore, the simulation in the dissertation is packet-level.

2.10.3 Validation of the Protocols in the Experiment

The packet-level simulation has been compared with the real-world experiment in [145]. The result shows the Constant-bit-rate (CBR) data traffic from ns-2 is realistic compared to that from the real-world experiment.

The AODV-UU package for ns-2 is from Uppsala University, which is based on the latest AODV draft. AODV-UU was tested in both the real experiment environment and ns-2, and the test results are similar. The detailed document as well as the discussion is in the designer's Master's thesis [151]. The simulation in the dissertation is using AODV-UU as the routing protocol. AODV-UU is also verified and validated by the authors and other researchers [152-154]. The author of AODV-UU used verification methods of model checking and deductive verification in [152]. Chakeres and Belding-Royer verified the AODV with a four node network and a five node network before they implement their proposed architecture [153]. Musuvathi, Park, et al, proposed a new C

model checker (CMC), one model they were doing the experiment on is the AODV-UU, and the result shows AODV-UU has good performance in the properties checking and the event handlers checking, and has higher correctness specification in the code checking compared to other AODV implementations [154].

In this research, a large numbers of tests were done on the ns-2 and AODV-UU. The comparison was done among the original network system (referred as Peace runs), the network under attacks (Dataflooding runs), the network applied with the defense system (Shield runs), and the network applied with the defense system under attacks (Defense runs). The test result shows the peace runs and the dataflooding runs are match the protocol and network specifications.

2.11 Summary

Mobile Ad-hoc Networks can be applied in all kinds of scenarios, including school educational environments, military fields, civilian communities, factory plants, and many more. But the development of the hardware infrastructure and the networking software, especially the security protection, is not meeting the demand. Some traditional network security shortcomings and attacks are not solved. On the contrary, because the MANETs are more vulnerable than wired networks, the security attacks become much more severe threats.

Analysis and experiment results show an appropriate MANET routing protocol should be reactive, anonymous and stateless. AODV is an outstanding MANET routing protocol that satisfies these requirements. But it has no security defense. Therefore, a practical and effective security solution is needed for AODV to protect the networks from

the security threatens. That will be the last step and the vital step before the AODV based system is put into operation in the risky real world.

Up to the present, all MANET security strategies need modifications on either network protocols or network topology, which may be not feasible for an operating commercial MANET. Even a current state of the art protocol or a successfully updated protocol may face DDoS attacks armed with newer technologies in the future. Therefore, a new practical problem approach is demanded to defense MANETs against DDoS attacks.

The proposed secure strategy is running on a MANET with AODV or AODV-like routing protocols. The experiment will be implemented in the ns-2 network simulation environment because ns-2 simulator is widely applied and it is validated and verified.

CHAPTER 3

DETAILED DESCRIPTION OF THE RESEARCH

3.1 Assumed Environment

The dissertation focuses on operating AODV-based MANETs. Each unit is an independent wireless mobile node that has both networking and computing capacity. Each node is designed to comply with the AODV protocol [155], and cooperate on routing and data forwarding. Each node arbitrarily joins and leaves the network, and the nodes and the wireless connections are fragile and unstable. So the topology of the network is variable. The DDoS attackers could be any combination of nodes, even including some compromised security system nodes. The dissertation focuses on the DDoS attacks of flooding and black-hole.

3.2 Definition of the Problem

An operating AODV-based MANET would require a security strategy to defend against existing and potential DDoS attacks. Because the MANETs are in a dispersal pattern, and the nodes may be individually controlled, some or all nodes are out of reach or even out of control of the network administrators. It is difficult to apply a network wide security upgrade. Not only operating MANETs but also any upcoming or planned MANET would meet this problem. Though an about to be deployed MANET can apply

an updated defense strategy, any unpredictable, unforeseen DDoS attack technique in the future can threaten the network and put it in the same situation of those operating unsafe MANETs.

A proper solution should require little or no change of the existing network system. It should not require a large-scale upgrade. It should not depend on cooperation from the individuals. It should be backward compatible and transparent to the other components of the network.

3.3 Design Principles Used by Proposed Solution

Based on my survey and analysis of current MANET DDoS attacks and defense strategies, I conclude the fundamental principles of designing the secure MANET routing protocols and the DDoS defense strategies, which should also apply on AODV defense.

- The number one goal: stop evil, protect the innocent. The DDoS attacks abuse the definition of the network protocols by consuming network or system resources, so that other legitimate requests cannot be served [156]. A good DDoS defense should reactively stop only the malicious traffic, and ensure that legitimate traffic is passed. There are two opposing methods of defense. One is to reinforce the tolerance and resilience capability of the system [157] by applying encryption protection [134, 158], evading the attacks [159, 160], allowing redundancy [161]. On the other hand, significant efforts actively stop the attacks by monitoring and filtering the malicious traffic, and/or tracing back to the attackers, such as Mayday [135], Hop-Count Filtering [162], and Hash-based trackback [39]. The methods under this category

require large-scale deployment of the defense routers and a necessary modification on the network layer protocols, but they attempt to solve the specific problems effectively from the source.

- Distributed detecting architecture efficiently responds to attack. The defense initiated by a victim is much less practical or efficient. Because of the nature of the attack traffic, the overwhelming traffic could have already congested and crashed the intermediate nodes before it reaches the victim (Refer to Section 2.3 Distributed DoS (DDoS) Attacks in Wired Networks and the Internet). Victim initiated intrusion detection could not meet the attack traffic and fail in this scenario. So an efficient detection and defense strategy should be based on a distributed architecture and is initiated by the intermediate nodes.
- The defense system should be dynamic, distributed, adaptable, and effective. Dynamic means a filter or a defense node can join and exit as needed [134]. There should be no constraint on the deployed topology; distributiveness means the security workload is shared and balanced among the defense system nodes; adaptability means the defense system does self-recovery on the network change or under the DDoS attacks; effectiveness requires the system to be functional under DDoS attacks, which includes detecting, logging and reporting, tracing back the attack path, isolating the attackers and ultimately, stopping the attack. There are many external restrictions for a security system to fulfill all the functionalities listed above. Some

functionalities are infeasible for some specific network architectures or under some specific scenarios.

- If a defense system is required for an operational network, the practicality of the strategy will be a crucial issue. In this situation, the system should try to avoid modifying the current network infrastructure, try to impact as fewer nodes as possible, and try to build the anti-DDoS in the application layer of the participating routers and nodes. Less modification means less cost, or more practicality to the public or commercial networks.
- The defense system should work for all types of network traffic, which means it can fully control the target's bandwidth (in contrast to the strategy that works only on a type of traffic, such as a web browser service [163]), and for all types of customer, which agree to the model of a public service, such as Yahoo.com (in contrast to the strategy that needs legitimate customers before building the link [112, 127, 128, 134]). The defense system requires generalization of the system to fit in the practical environment and defend against generic DDoS traffic.
- Tracking down a specific packet or source is not very useful in defending against a DDoS attack because many attack sources are only randomly compromised nodes. The tracking procedure consumes resources too. This is especially true in MANETs, where either the links or the location of nodes is volatile. The only good way to track back the malicious traffic is to have a guard located as close to the attacker as possible [134].

- The defense system should provide good performance with the least overhead. QoS capability is required for some commercial scenarios. A defense system may protect only one specific network service. Several individual defense systems may work together to provide an overlay-based protection. The owner of the defense system can be the ISP, a public security organization, or a security service provider. Therefore, different scenario may have different QoS demands.
- The defense system should assume that the defense mechanism and structure may be fully known by all network members including the attackers, and still have self-protection. So it should be durable toward attacks on the system itself. Some protection tactics should be taken. For instance, the system should be able to detect suspicious or malicious scanning from the attackers and return phony or misleading information, or evade the attack by address hopping [139].

3.4 Proposed Resolution

3.4.1 Basic System Architecture

This dissertation proposes a proxy-based security system. The protected target is a specific service, located on a service provider S, and delivered over several proxy nodes, where each node is called guard G. The service is described as a tuple (SVR, ip), where the SVR is the service name, and the ip is the artificial IP address announced for the service. A service requester, node R, broadcasts an AODV routing lookup packet RREQ of ip. The guard nodes respond to the RREQ with a RREP to announce an available

route to the ip. R may receive several RREP from the guard nodes, and it keeps the shortest route, for example, through G_i, according to the AODV protocol. G_i forwards the data packets of R to S by tunneling, and forwards the data packets from S back to R in a normal data packet (Figure 23). G maintains the connection to the S. G nodes have knowledge about the overlay system and periodically exchange the information among each other. S randomly sends out a RREQ for ip to discover any possible black-hole attackers.

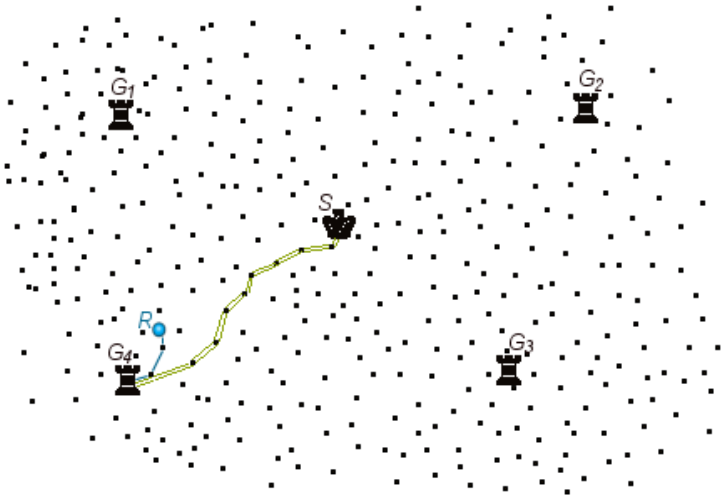


Figure 23. Routing in the proposed example MANET. S is the Service Provider, G_i (i = 1 ~ 4) are the Guard Nodes. R is the Service Requester. Blue path is R's RREQ and Service Request route. Green path is an encrypted data tunnel.

3.4.2 Attack Scenarios

The number of RREQ packets a router can generate or forward in one second is defined by the RREQ_RATELIMIT. The value of this configuration specified in the AODV standard RFC3561 is 10 (packets per second). If a node has transmitted, (either by its own or forwarded), 10 RREQ packets, it will drop the subsequent RREQ packets in a one second interval. When the second is up, the node will resume the transmission up

to the rate limit. A RREQ flooding attacker may disable the rate limit for sending as many RREQ packets as it can. But the intermediate nodes will only forward up to 10 RREQ packets. Although this system setting prevents the RREQ attack flooding from continuously saturating the network, the attack transmission peak of each second nevertheless uses up the RREQ ration quickly, thereafter, all the legitimate RREQ packets are dropped too, and the whole network fails to build routes as long as the attacks are ongoing.

In AODV a node looks up the incoming RREQ in its routing table, and if it has any matched entry, it will not forward the request, and respond with a RREP packet. A flooding attacker can use a non-existent IP as the destination to have the RREQ packets forwarded throughout the network. It will result in the DoS flooding of the network functionality of routing discovery as described above. Because RREQ flooding attacks are limited by the RREQ_RATELIMIT according to the AODV definition, so such attacks cannot take up the bandwidth all the time, meanwhile the data packet rate is not limited by the RREQ_RATELIMIT, and the transmission among defense system nodes is based on data packets, so RREQ flooding attacks can not harm the defense system operating. Routing information can be tunneled as data packets and exchanged among defense system nodes.

When a RREQ flooding attack targets the service provider S, either the attack traffic is directed towards several guard nodes, or the whole attack traffic is forced on the region of one guard node, e.g. G_i, which could be crashed by the attack. But more likely, according to the AODV definition, the nodes routed to ip/G_i respond with the route by G_i.

The RREQ attack traffic stops inside the subset of region of G_i, the rest of the network and overlay system continue to operate (Figure 24).

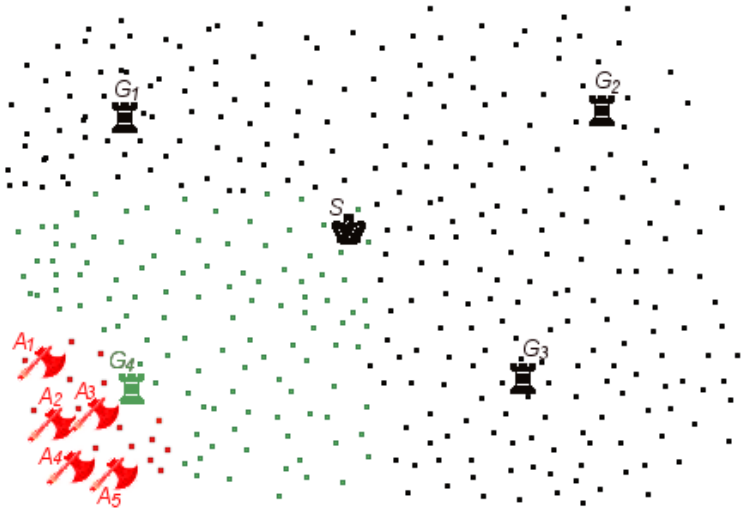


Figure 24. Illustration of a RREQ flooding DDoS attack on service provider S. The green region has the route to ip by G_4. The DDoS attack by the attackers A_i (i = 1~ 5) is restricted in the red region because these RREQ will be respond with a RREP by the red nodes.

When a data-flooding attack is launched, and the attack traffic is put upon a guard node (G_i), G_i can notice the suddenly increased request traffic so that it may take proper application level reaction, which depends on the service content. But it can also conservatively filter the incoming requests, which is described in Chapter 3, Section 3.4.3.3. The second defense method is to increase the forwarding rate moderately, to snatch a share of the bandwidth from the malicious traffic, which is discussed in Chapter 3, Section 3.4.3.4. The last defense method is to keep recovering the false negative broken routes from the guards to the service provider under the attack, which is described in Chapter 3, Section 3.4.3.5.

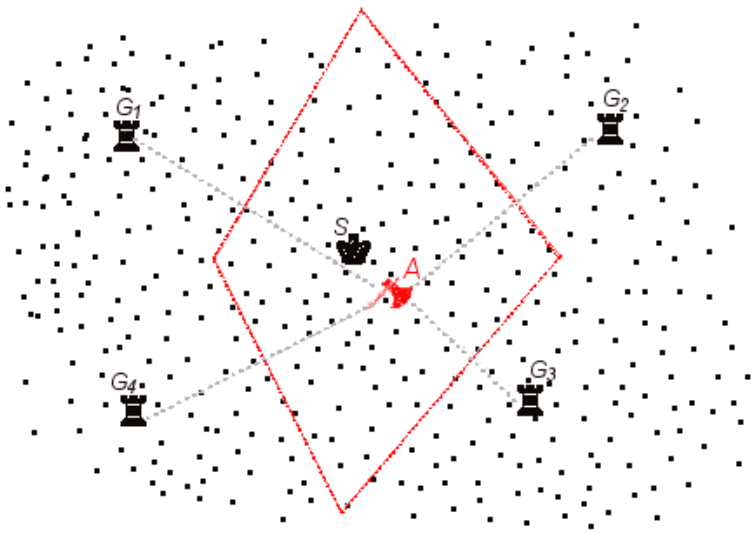


Figure 25. Black-hole attack scenario 1. The attacker A is close to S. The nodes in the red region are closer to A than any G.

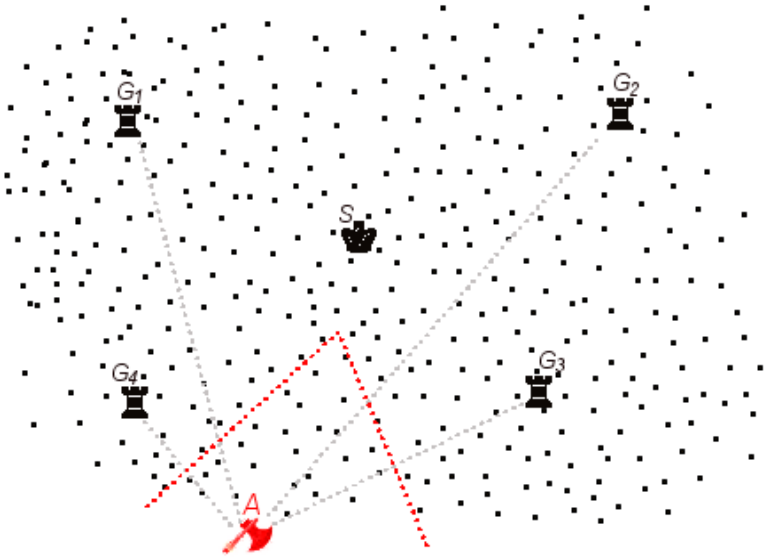


Figure 26. Black-hole attack scenario 2. The attacker A is close to an edge. The nodes in the red region are closer to A than any G.

When a black-hole attack is lunched, the RREQ packets are responded to by the attacker A with the spoofed route to S. Figure 25 and Figure 26 illustrate the two scenarios of the black-hole attacks. The detailed defensive mechanism is described in Chapter 3, Section 3.4.3.6.

Because the guard nodes are the essential entry points of the defense system, they must be strictly protected from exposure attacks or penetration attacks, especially when the attackers have knowledge of the existence and mechanism of the defense system. A penetration attacker may send a RREQ request, analyze the RREP responses, and find out which neighbor node has fewer hops to the service provider, move toward that direction and repeat whole process, until eventually a node is found the shortest path, and it may conclude that this node is a guard node. One solution is to change the IP address and position of the guard node periodically to avoid tracking. The detailed defensive mechanism is described in Chapter 3, Section 3.4.3.2. In case a penetration attacker reveals a guard node, it may eavesdrop and analyze the outgoing traffic from a guard node to discover the server's real IP address. Besides periodically shifting the guard node, each guard node uses two different IP addresses for inner and outer system communication to increase the concealment. The detailed defensive mechanism is described in Chapter 3, Section 3.4.3.7.

3.4.3 Advanced Defense Mechanisms

3.4.3.1 Encryption

S and G use encrypted tunnels to transfer the payload. It increases the computational complexity at the ends of the tunnels. The encryption algorithm used by the system is Elliptic Curve Digital Signature Algorithm (EC-DSA; as specified in ANSI X9.62) [164, 165] that is regarded as a more efficient public key algorithm for MANET and sensor networks [166]. The algorithm codes in the experiment are from the cryptographic toolkit by libtomcrypt.org.

3.4.3.2 Shift of Guard nodes

Guards G periodically change their physical IP address and position. Only S has synchronized information of G. Every time a G shifts, all the nodes it served must rediscover routes to the service provider. If a penetration attacker tries to reveal a guard node by tracking along RREPs with a lower number of hops, it will fail when the guard shifts and the routes altered. This mechanism protects guard nodes from malicious scanning, eavesdropping, and the penetration attacks. The system overhead increases by the periodical routing rediscovery by about 3 times in 900 seconds in the experiment simulation (Appendix C.4). The model and implementation of the route tracing attack are discussed in Appendix C.

3.4.3.3 Self-adaptive and Dynamic Reconfiguration

G nodes manage a privilege list of the service requester R nodes, which can be a given list by the server S, or simply a list of the served nodes. When a G is under a flooding attack, it will keep up the connection towards those on the privilege list, and drop the newcomer. A further assurance is to share the list among the neighbor G nodes. When a guard node G_i crashes, its neighbors can accept and respond to the new RREQ from the privileged Rs. On the other hand, when a node R lose the route to S because of the crash of G_i, it can resend the RREQ, and the request can be responded by the next available G_j, therefore the route from R to S is detoured and rebuilt (Figure 27).

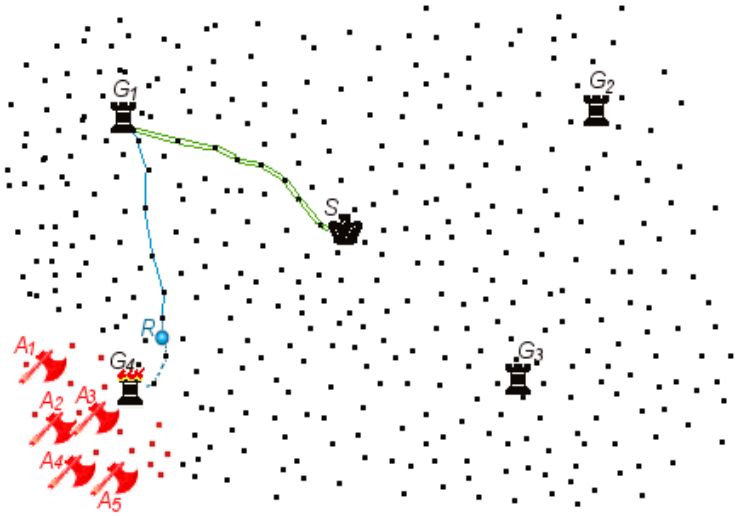


Figure 27. Illustration of a DDoS data flooding attack. G_4 was crashed by the attack. R lost the route of G_4. It resends RREQ, and nearby guard node G_1 responds, and then a detour route is built.

3.4.3.4 Raise the Transmission Speed Moderately

Based on the characteristics of the wireless networks and AODV routing protocol, the Guard nodes raise the transmission speed moderately when the network is under a flooding attack. This flexibility lets the defense system snatch more bandwidth from the malicious traffic. The malicious transmission cannot saturate the time slots, so more transmission attempts may let a guard obtain more available slot. The experiment result shows the method may improve the delivery rate significantly under some circumstances. But a raise of the transmission rate often aggravates the traffic congestion upon the flooding attack. The experiment result shows there is a peak delivery rate a legitimate traffic can technically reach in a specific scenario.

The same “sweet spot” effect rules the flooding traffic too. The maximum transmission speed does not necessarily lead to the most harmful network impact. The continuous radio collision can congest the attack traffic itself at the immediate neighbors

of an attacker and stop the attack traffic going beyond the attacker's radio frequency region.

3.4.3.5 Forced Rapid Route Recovery

When the network is under DDoS data flooding attack, the routes are frequently lost by the traffic congestion. The temporary link non-availability is treated as a common link broken by the nodes and the data packet is dropped as defined in AODV protocol. When a guard node receives a RERR, it will immediately send another AODV route discovery to restore the route to the service provider. My experiment result shows the method of forced rapid routing recovery will improve the delivery rate up to double.

3.4.3.6 Dynamic Announced Hop Number

G nodes periodically, synchronously announce a random hop number (h) in RREP packets. h is the artificial distance away from the service provider. Because the G nodes are supposed to be the nearest ones to the service, technically h can be any in a range of 1 to (TTL-R_G) where constant R_G is the radiator of the region of a G in (Figure 24). S periodically sends RREQ for ip. If the received RREP shows a hop number that is abnormally shorter than ($h + R_G$), there could be a black-hole attacker in the route. If the attacker attempts to spoof a larger hop number, but the h from G could be smaller at the moment, the attack fails.

3.4.3.7 Dual IP Concealment on Guard Nodes

To have better protection from penetration attacks, each guard node uses two different IP addresses for inner and outer system communication. The guard replies with the service requesters with the address ip_{outer}, while it forwards the enciphered data to the server with another address ip_{inner}. ip_{inner} does not participate in routing discovery, so it is not released to the outside of the security system. The double-role guard behaves as there are two different nodes. The method protects the security system from the basic penetration attacks. But an attacker may eavesdrop and analyze a great quantity of traffic packets to discover the ip_{inner}. How easily a guard node can be discovered depends on the network topology, the background traffic status and the attack methods. If the tunneled traffic is extraordinary, more IP-pairs are required to split the traffic into normal-sized streams, i.e. to use more ip_{inner} address on the guard side, and the same number of IP on the server side. The backward stream needs forwarded by the guard nodes too.

3.5 Summary

This dissertation proposes a proxy-based security system. It focuses on the flooding and black-hole DDoS attacks on the existing operating MANETs and the practical solution for the problem. The system is self-adaptive and it protects specific service provider nodes from the DDoS attack traffic. The system can protect itself from penetration attacks by concealing the guard nodes and their traffic. The defense mechanisms used in the research are:

- Proxy based overlay architecture.
- Client priority management.

- Traffic management application.
- EC-DSA encryption.
- Guard nodes periodically change physical IP and position.
- Self-adaptive and dynamic reconfiguration.
- Forced rapid route recovery.
- Dynamic announced hop number by guard nodes.
- Dual IP concealment on guard nodes.

CHAPTER 4

SIMULATION AND EXPERIMENT DESIGN

4.1 Introduction

The goal of the experiments in the dissertation is to illustrate and prove the capability and the feasibility of the proposed defense system. The defense system is implemented and tested in the network simulator ns-2. A simulated network of 50 randomly generated and uniformly distributed wireless nodes is used for each stage of the experiment. Each stage introduces one or a set of factors to the fundamental network (Table 1, Table 2 and Table 3).

Table 1. Experiment 1, Data Flooding Attack at Server

Stage	Section	Runs	Description
1	4.5.1	Normal Operation	Bare MANET with AODV routing and CBR traffic
2	4.5.3	Data Flooding Attack at Server	Data flooding attack at Server on Normal Operation run in 4.5.1
3	4.5.8	Proposed Security System	Proposed security system works on Normal Operation run in 4.5.1
4	4.5.10	Proposed Defense Response to Data Flooding Attack at Server	Proposed security system response on Data flooding attack at server run in 4.5.3

Table 2. Experiment 2, Data Flooding Attack at Requester

Stage	Section	Runs	Description
1	4.5.1	Normal Operation	Bare MANET with AODV routing and CBR traffic
2	4.5.4	Data Flooding Attack at Requester	Data flooding attack at requester on Normal Operation run in 4.5.1
3	4.5.8	Proposed Security System	Proposed security system works on Normal Operation run in 4.5.1
4	4.5.11	Proposed Defense Response to Data Flooding Attack at Requester	Proposed security system response on Data flooding attack at requester run in 4.5.4

Table 3. Experiment 3, Random Data Flooding

Stage	Section	Runs	Description
1	4.5.1	Normal Operation	Bare MANET with AODV routing and CBR traffic
2	4.5.5	Random Data Flooding Attack	Random data flooding attack on Normal Operation run in 4.5.1
3	4.5.8	Proposed Security System	Proposed security system works on Normal Operation run in 4.5.1
4	4.5.12	Proposed Defense Response to Random Data Flooding Attack	Proposed security system response on Random data flooding attack run in 4.5.5

First, the simulator network runs the AODV routing protocol. A set of random and moderate Constant-bit-rate (CBR) traffic is generated to simulate a normal network workload. The generic networking metrics are logged and referred to as the baseline of the normal network behaviors. Second, three DDoS attacks: RREQ flooding, data flooding, and Black-hole, are implemented and injected individually into the network

during Stage One. The impact and damage are recorded respectively. Third, the proposed defense system is implemented and launched in the network at Stage One. The network overhead introduced by the defense system is measured. Fourth, the defense system from Stage Three confronts each DDoS attack from Stage Two. The network metrics are recorded and will be compared to the data from Stage Two.

4.2 Simulation Scenario

The simulation parameters are the average values from other typical experiments [64, 137, 167]. For example, the simulation network has 50 wireless nodes randomly scattered on a flat surface of 1500 meters by 500 meters. This network size is moderate for efficient AODV networking, because AODV has problems supporting large scale MANETs [168]. Each node has routing and computing capacity, and it communicates with its neighbor nodes by a radio wave with the range of 250 meters (Figure 28) and 802.11b as the MAC layer protocol. All nodes use the same mobility model. The maximum node speed is 50 meters per second, i.e. 111.87 miles per hour, which can be considered as the top speed a mobile node can reach. All nodes including DDoS attackers and defense system nodes are equal in hardware and computation capacity. When the network is under DDoS data flooding attack, the guard nodes allow only the legitimate traffic from two known requesters, but all the rest of the traffic will be dropped. The detailed specifications are listed in Table 4.

Table 4. Simulation Parameters

Dimension	1500 m × 500 m
Node number	50
MAC protocol	IEEE 802.11b
Radio frequency (RF) range	250 m
Routing protocol	AODV
Workload traffic type	Constant-bit-rate (CBR)
Packet size	512 Bytes
Background traffic rate	1 packets per second (p/s)
Source number	20
Traffic Start Time	0
Mobility	Up to 50 m/s
Simulated period	900 seconds
Service Clients	2 nodes
Packet size	512 Bytes
Packet sending rate	10 p/s
Service Server	1 node
DDoS Flooding Attack Nodes	5 nodes
Data Flooding Packet size	1024 Bytes
Packet sending rate	40 p/s
Attack Start Time	1
Defense System	
Guards	Up to 4 nodes
Processing Delay on Guard	Machine processing time between recv() and send() of a packet
Guard normal rate	10 p/s
Guard under-attack rate	Up to 20 p/s

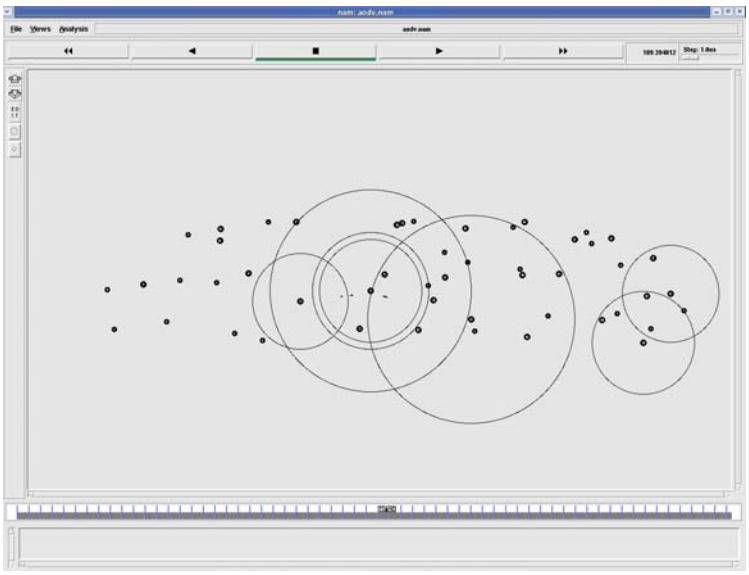


Figure 28. Illustration of the Simulation Topology

4.3 Simulation Environment

The computer used for the experiment is a PC with an Intel Celeron 1.80GHz CPU and 1GB of RAM. The operating system is Redhat Fedora 2.0 with Linux kernel 2.6.10. The experiments are implemented and run in the network simulator ns-2 (version of 2.29). Instead of using the Carnegie Mellon University AODV packet, which comes along the ns-2 bundle, the experiments use AODV-UU (latest version 0.9.1) from Uppsala University as the routing protocol. The advantages of the AODV-UU packet are that not only are its features up-to-date, but AODV-UU has been verified and proven against real-world experiments. Details of AODV-UU are in Chapter 2, Section 9.3. The implementation of the dissertation modifies the AODV-UU source code to add both DDoS and defense system features. A broadcast flag bug in AODV-UU-0.9.1 was found and reported to the author team of AODV-UU on March 16, 2006.

Source code files changed or added are: (Appendix A)

/aodv_rreq.c

/aodv_socket.c

/ns/adov-shield.cc

/ns/aodv-uu.cc

/ns/packet_input.cc

The test program is a tcl script file.

4.4 Experiment Metrics

1. End-to-end latency.

End-to-end latency describes the time taken for transmitting a packet from the source to the sink. The experiment measures end-to-end latency to demonstrate when a network node or section is congested or under DDoS attack.

2. Network throughput and packet delivery rate.

The network throughput is the amount of data moved successfully from one place to another in a given time period. It is the most significant measurement and indicator of the experiment. The goal of the proposed defense system is to increase the legitimate network throughput under DDoS attacks.

The packet delivery rate is defined as

$$R = \frac{\text{Packets Sent by Sources}}{\text{Packets Received by Sinks}}$$

And it is another aspect of the network throughput.

The packet sending rate of the service requesters is 10 p/s (40kbps) representing normal audio traffic. The delivery rate in the experiment is measured as the packet number received by the service provider in each second.

3. The number of overall packets dropped, the number of legitimate packets dropped, and the number of malicious packets dropped.

The number of overall packets dropped describes the network congestion.

The number of legitimate packets dropped describes the network congestion and the system vulnerability to DDoS attacks. The proposed defense system attempts to lower the number of legitimate packets dropped.

The number of malicious packets dropped describes the capacity of the defense system. The proposed defense system attempts to raise the number of malicious packets dropped.

4. System overhead.

It is the total number of the defense system maintenance packets. System overhead describes the deployment cost of the system.

4.5 Experiment Schemes

4.5.1 Normal Operation

Run the network described in Chapter 4, Section 2. Start a service provider on node 35 who is conveniently near the center of the network, and two service requesters, node 21 and node 30. The service requester nodes communicate to the service server node by CBR streams (Figure 29).

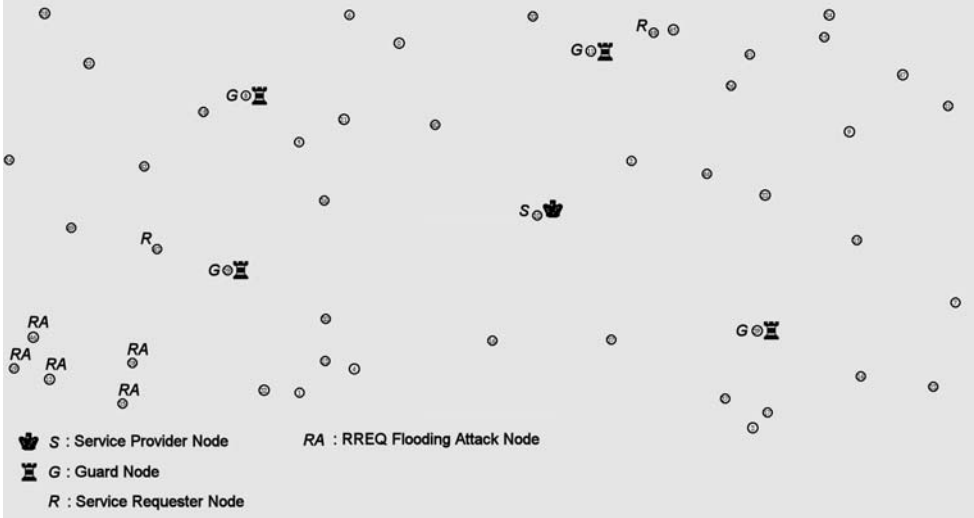


Figure 29. Illustration of the Experiment Setting for the Experiments of Normal Operation, RREQ Flooding Attacks, Proposed Security System and Proposed Defense Response to RREQ Flooding Attacks.

Data collection plan:

- Record the network throughput, which will be the baseline for the normal network workload.
- Record the end-to-end latency from R to S, which will be the baseline for the service traffic of a normal operating network.
- Record the delivery rate from node 21 to S, from node 30 to S, and overall delivery rate, which will be the baseline for the service capacity of a normal operating network.
- Record the overall packet drop rate and overall legitimate packet drop rate, which will be the baseline for the network capacity of a normal operating network.

4.5.2 RREQ Flooding Attack

Run the simulation under operating conditions. Launch five malicious nodes 22, 0, 33, 28 and 49 to initiate the RREQ flooding (Figure 29).

Data collection plan:

- Record the network throughput and the end-to-end latency from R to S. Compare them against the results from normal operation.
- Record the number of overall packets dropped and the number of legitimate packets dropped.

4.5.3 Data Flooding Attack at Server

Run the simulation under operating conditions. Launch five malicious nodes 41, 39, 12, 27 and 44 to send the CBR flooding to S (Figure 30). The malicious nodes are put around the server node to have a direct attack on the server, and do not need packet forwarding by other nodes. The scenario can be considered as the worst-case scenario for the server.

Data collection plan:

- Record the network throughput, packet delivery rate and the end-to-end latency from R to S. Compare them against the results from normal operation.
- Record the number of overall packets dropped, the number of malicious packets dropped and the number of legitimate packets dropped.

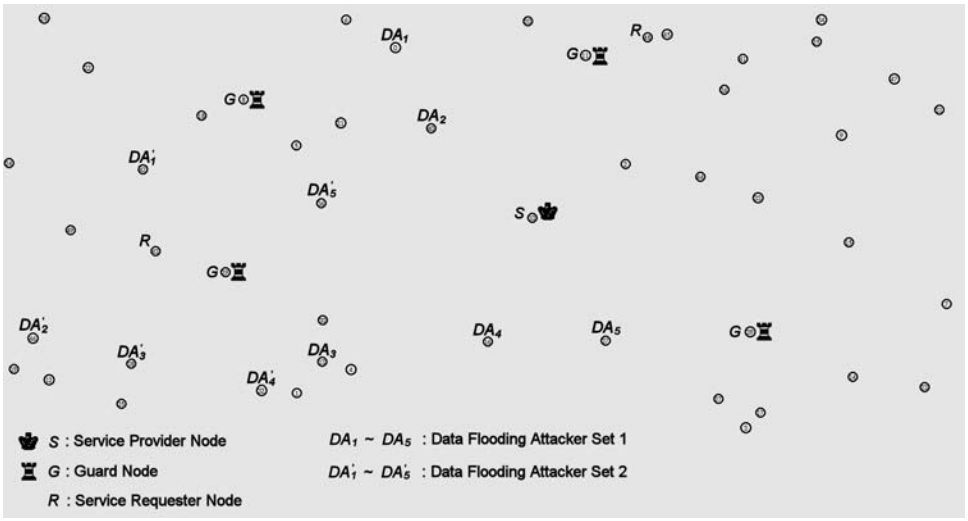


Figure 30. Illustration of the Experiment Setting for Data Flooding Attack at Server, Data Flooding Attack at Requester, Proposed Defense Response to Data Flooding Attack at Server and Proposed Defense Response to Data Flooding Attack at Requester

4.5.4 Data Flooding Attack at Requester

Run the simulation under operating conditions. Launch five malicious nodes 28, 18, 43, 1 and 6 to send the CBR flooding to S (Figure 30). The malicious nodes are put around a service requester node 21 to have a direct attack on it. The scenario can be considered as the worst-case scenario to the service requester node 21.

Data collection plan:

- Record the network throughput, packet delivery rate and the end-to-end latency from R to S. Compare them against the results from normal operation.
- Record the number of overall packets dropped, the number of malicious packets dropped and the number of legitimate packets dropped.

4.5.5 Random Data Flooding Attack

Run the simulation under operating conditions. Launch five randomly selected malicious nodes 22, 0, 33, 28 and 49 to send the CBR flooding to S. The malicious nodes have full-random and unpredictable mobility generated by cmu-scen-gen application (Carnegie Mellon Scene Generator). The scenario can be considered as a generic attack situation.

Data collection plan:

- Record the network throughput, packet delivery rate and the end-to-end latency from R to S. Compare them against the results from normal operation.
- Record the number of overall packets dropped, the number of malicious packets dropped and the number of legitimate packets dropped.

4.5.6 Black-hole Attack 1

Run the simulation under operating conditions. Launch one malicious node 32 which is close to S as illustrated in figure 25 to answer all RREQ(ip) before service requester R has the route to service provider SVR.

Data collection plan:

- Record the end-to-end latency from R to SVR. R is expected to lose connection to SVR.

4.5.7 Black-hole Attack 2

Run the scheme of black-hole attack 1 except the malicious node 2 which is away from S as illustrated in figure 26.

Data collection plan:

- Record the end-to-end latency from R to SVR. R is expected to lose connection to SVR.

4.5.8 Proposed Security System

Run the simulation under operating conditions. Start the defense system nodes, which are one service provider S on node 49, four service guard nodes 31, 9, 38 and 10, and two service requesters, node 30 and node 21. They communicate to the service provider by CBR stream (Figure 29 and Figure 30).

Data collection plan:

- Record the network throughput, which will be the baseline of the normal network workload.
- Record the end-to-end latency from R to S, which will be the baseline of the normal service traffic.
- Record the number of overall packets dropped and the number of legitimate packets dropped.

4.5.9 Proposed Defense System Response to RREQ Flooding Attack

Run the proposed security system scheme. Launch the same five malicious nodes of RREQ flooding attack.

Data collection plan:

- Record the network throughput.
- Record the end-to-end latency from R to S. Compare them against the results from the RREQ flooding attack and proposed security system.
- Record the number of overall packets dropped, the number of legitimate packets dropped, and the number of malicious packets dropped.

4.5.10 Proposed Defense System Response to Data Flooding Attack at Server

Run the proposed security system scheme. Launch the same five malicious data flooding attack nodes on server.

Data collection plan:

- Record the network throughput.
- Record the end-to-end latency from nodes R to S. Compare them against the results from normal operation, data flooding attack at server and proposed security system.
- Record the overall and individual delivery rate from nodes R to S. Compare them against the results from normal operation, data flooding attack at server and proposed security system.
- Record the number of overall packets dropped, the number of legitimate packets dropped and the number of malicious packets dropped.

4.5.11 Proposed Defense System Response to Data Flooding Attack at Requester

Run the proposed security system scheme. Launch the same five malicious data flooding attack nodes at requester.

Data collection plan:

- Record the network throughput.
- Record the end-to-end latency from R to S. Compare them against the results from normal operation, data flooding attack at requester and proposed security system.
- Record the overall and individual delivery rate from nodes R to S. Compare them against the results from normal operation, data flooding attack at server and proposed security system.
- Record the number of overall packets dropped, the number of legitimate packets dropped and the number of malicious packets dropped.

4.5.12 Proposed Defense System Response to Random Data Flooding Attack

Run the proposed security system scheme. Launch the same five malicious random data flooding attack nodes.

Data collection plan:

- Record the network throughput.
- Record the end-to-end latency from R to S. Compare them against the results from normal operation, data flooding attack at requester and proposed security system.

- Record the overall and individual delivery rate from nodes R to S. Compare them against the results from normal operation, data flooding attack at server and proposed security system.
- Record the number of overall packets dropped, the number of legitimate packets dropped and the number of malicious packets dropped.

4.5.13 Proposed Defense System Response to Black-hole Attack 1

Run the proposed security system scheme. Launch black-hole attack 1.

Data collection plan:

- Alarm when a black-hole attack is detected.

4.5.14 Proposed Defense System Response to Black-hole Attack 2

Run the proposed security system scheme. Launch black-hole attack 2.

Data collection plan:

- Alarm when a black-hole attack is detected.

4.6 Experiment Design Summary

The simulator used by this research is ns-2, one of the most popular network simulators. The test-bed settings are drawn from published researches [64, 137, 167]. Twelve experiment scenarios are designed to cover the testing scope and crosscheck the proposed security system.

The experiments focus on the metrics of network throughput, end-to-end delay, and number of packets dropped. These metrics are necessary evaluation indexes of the

performance of a network. The experiment results from the scenarios of normal operation, the attacked network without proposed defense system and the attacked network with proposed defense system are used to demonstrate the network functionality difference, and the network security performance improved by the proposed defense system.

CHAPTER 5

EXPERIMENT RESULTS

5.1 Defense System on RREQ Flooding Attack.

The AODV system configuration RREQ_RATELIMIT restricts the number of packets a node can send and forward in one second [169]. The system setting of this value is 10 (p/s). The setting on RREQ flooding attackers is assumed to be hacked to unlimited in this experiment by turning off one condition check on the RREQ count (Appendix A.3). But the intermediate nodes still forward the packets at the rate of RREQ_RATELIMIT, and any excess RREQ is dropped. This AODV system feature effectively smothers the AODV RREQ flooding attack as the attack moves through the network.

5.2 Defense System on Data Flooding Attacks.

5.2.1 Results Comparison of the Following Scenarios: Normal Operation, Data Flooding Attack at Server, Proposed Security System and Proposed Defense Response to Data Flooding Attack at Server

In this set of experiments, service provider is node 35, which is known only by the defense system nodes. The service is broadcasted with the pseudo service ID of 49 in the proposed security system, which is known by all nodes in the network. The guard

nodes of the defense system intercept the traffic to the service ID 49 and forward to service provider node 35. Service requesters are nodes 21 and 30. Five attackers are nodes 41, 39, 12, 27 and 44. During the data flooding attack scenarios, the attackers are deployed one hop away the server node, and sending data packets directly to the server. Once the simulation run starts, all nodes, include the server node, the proposed defense system nodes, service requester nodes, and attacker nodes are randomly roaming. This means at an arbitrary moment, a node is moving at a random speed and in a random direction.

5.2.1.1 Throughput

Table 5. Overall network throughput comparison for the following scenarios: normal operation, data flooding attack at server, proposed security system and proposed defense response to data flooding attack at server over 900 seconds.

Runs	Overall Throughput (MBytes)
Normal operation	7.79
Data flooding attack at server	1.89
Proposed security system	7.09
Proposed defense response to data flooding attack at server	4.00

By suppressing the attack traffic, the defense system helps the network gain more useful throughput under DDoS data flooding attacks occurring at the center of the network and targeted directly at the server. Table 5 shows the defense system improves the overall throughput by 111.64% when the server is under attack. The legitimate packets dropped before reaching the guard nodes are not rescued by the defense system.

5.2.1.2 Dropped Packets

The overall packet drop rate of the following scenarios: normal operation, data flooding attack at the server, proposed security system and proposed defense response to data flooding attack at server are illustrated in Figure 31. The overall legitimate packet drop rates are illustrated in Figure 32. The overall packet drop rates of the four scenarios are displayed in four colored lines. The rates of normal operation (in red[*]) and proposed security system (in green) are close to 0, but the rates of data flooding attack at server (in blue) and proposed defense response to data flooding attack at server (in brown) are close to 200 packets. The overall attacking packet drop rates are illustrated in Figure 33.

[*] Color convention is used by all graphics.

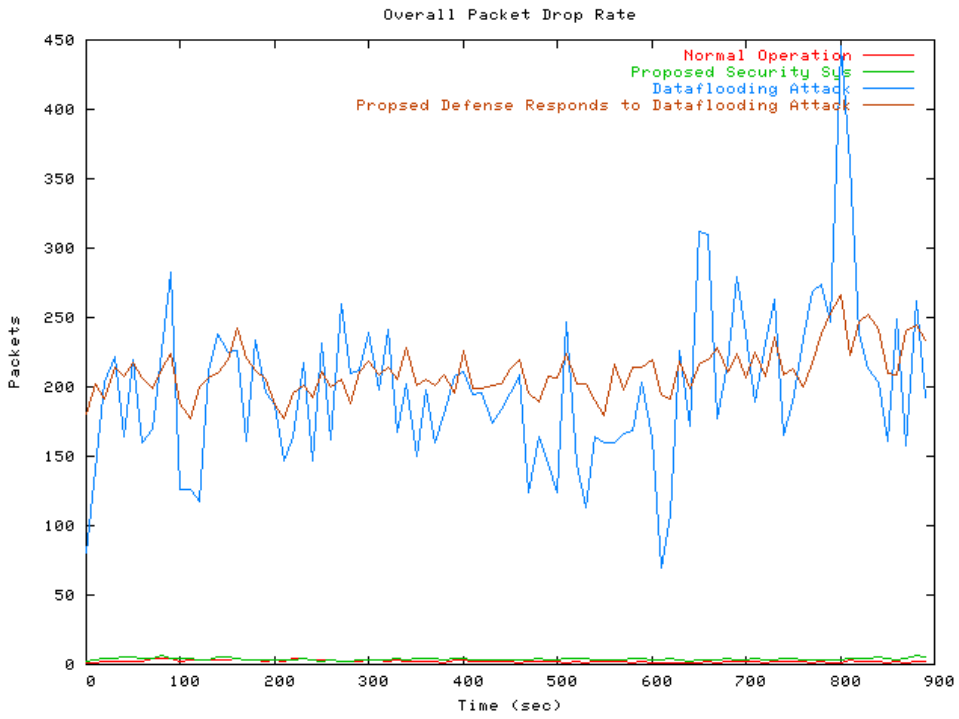


Figure 31. Overall Packet Drop Rate in Scenarios Normal Operation, Data Flooding Attack at Server, Proposed Security System and Proposed Defense Response to Data Flooding Attack at Server

Table 6. Overall drop rate comparison for the following scenarios: normal operation, data flooding attack at server, proposed security system and proposed defense response to data flooding attack at server

Runs	Overall drop rate (packets per second)
Normal operation	2.68
Data flooding attack at server	198.55
Proposed security system	2.75
Proposed defense response to data flooding attack at server	210.96

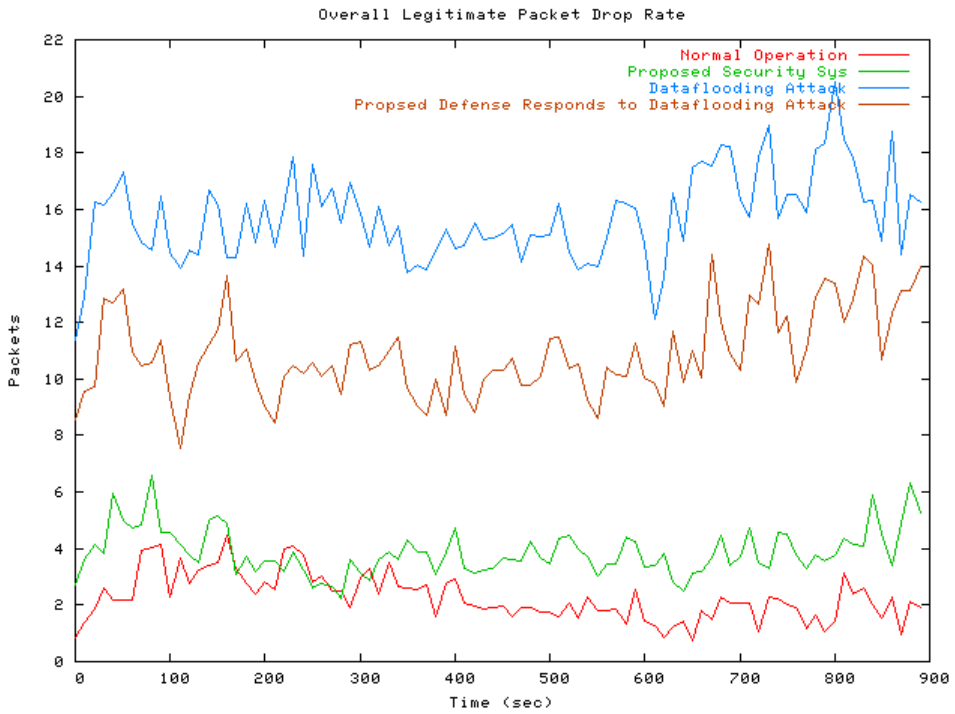


Figure 32. Overall Legitimate Packet Drop Rate in Scenarios Normal Operation, Data Flooding Attack at Server, Proposed Security System and Proposed Defense Response to Data Flooding Attack at Server

Table 7. Overall legitimate packet drop rate comparison for the scenarios: normal operation, data flooding attack at server, proposed security system and proposed defense response to data flooding attack at server

Runs	Overall legitimate packet drop rate (packets per second)
Normal operation	2.28
Data flooding attack at server	15.70
Proposed security system	3.87
Proposed defense response to data flooding attack at server	10.90

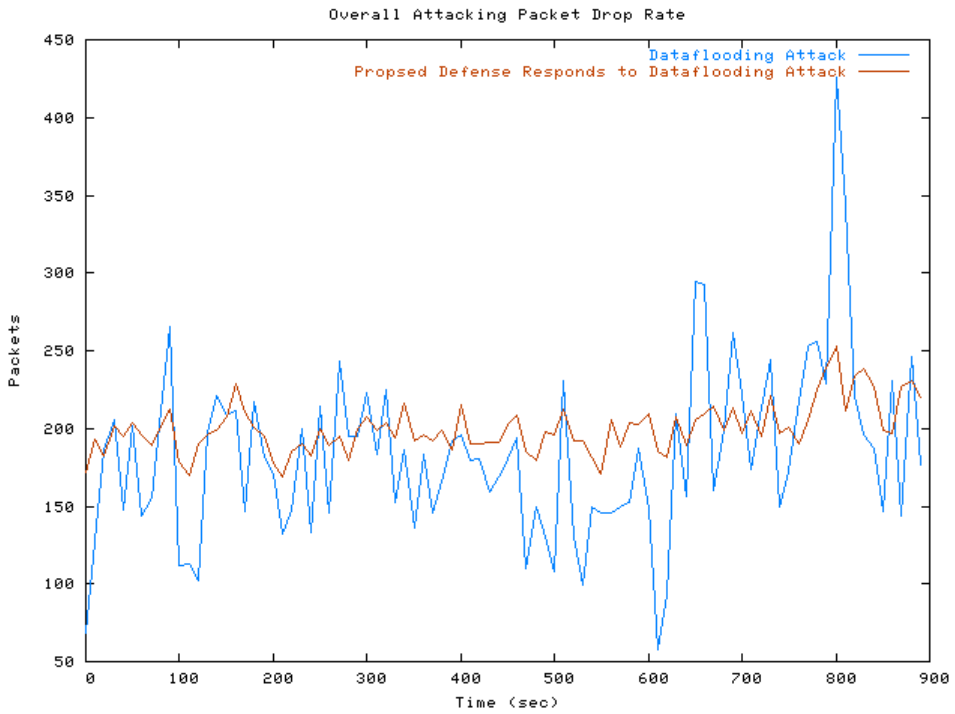


Figure 33. Overall Attacking Packet Drop Rate in Scenarios Data Flooding Attack at Server and Proposed Defense Response to Data Flooding Attack at Server

Table 8. Overall attacking packet drop rate comparison for the following scenarios: normal operation, data flooding attack at server, proposed security system and proposed defense response to data flooding attack at server

Runs	Overall attacking packet drop rate (packets per second)
Data flooding attack at server	182.85
Proposed defense response to data flooding attack at server	200.00

When the network is under a data flooding attack, Figure 31 and Table 6 show the defense system reduces the network traffic which is mostly attacking traffic; meanwhile, the defense system drops 30.57% less legitimate packets as shown in Figure 32 and Table 7. Most attacking packets dropped in an unprotected network are caused by naturally occurring congestion. But the defense system intentionally filters all those attacking

packets over the effect of attacking congestion. Figure 33 and Table 8 show there are no attacking packets going through the defense system.

5.2.1.3 Delivery Rate

The overall delivered packets per second from service requester node 21 in the following scenarios: normal operation, data flooding attack at server, proposed security system and proposed defense response to data flooding attack at server are illustrated in Figure 34. The delivered packets from service requester node 30 are illustrated in Figure 35. The brown line is overall higher than the blue line, which means there are more packets delivered in the network with the defense system than without the defense system.

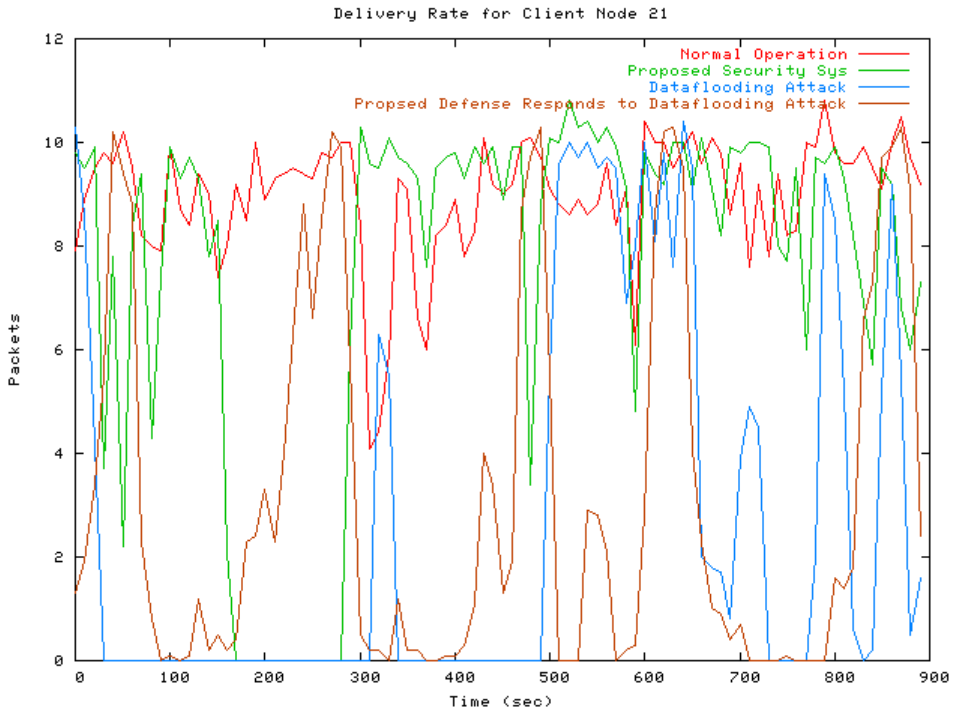


Figure 34. Delivered Packets for Node 21 in Scenarios Normal Operation, Data Flooding Attack at Server, Proposed Security System and Proposed Defense Response to Data Flooding Attack at Server

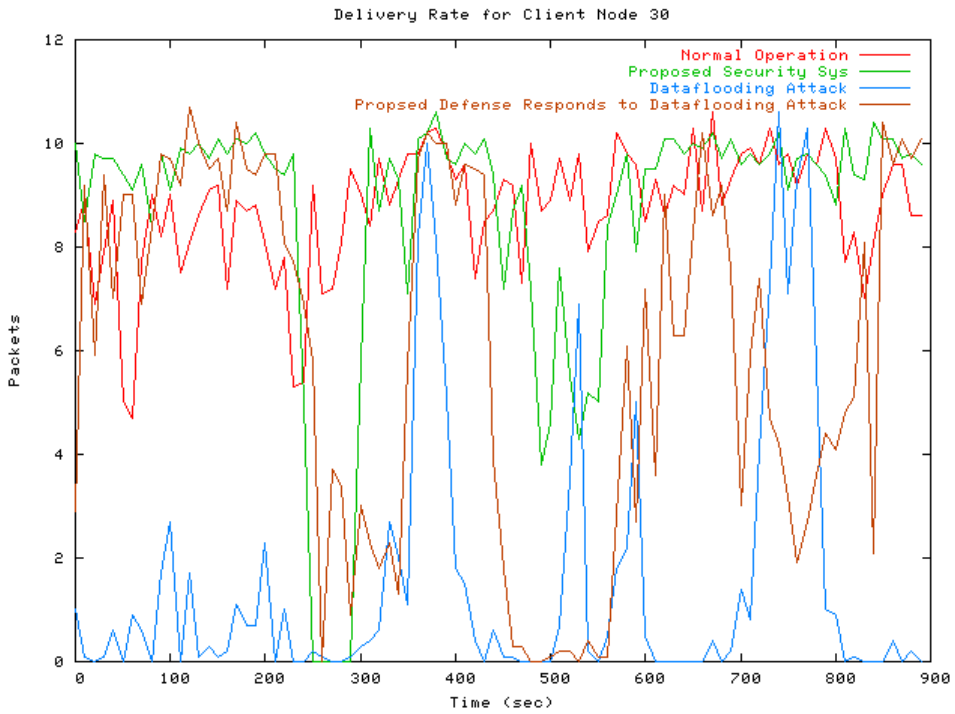


Figure 35. Delivered Packets for Node 30 in Scenarios Normal Operation, Data Flooding Attack at Server, Proposed Security System and Proposed Defense Response to Data Flooding Attack at Server

Table 9. Successfully delivered packets over 900 seconds comparison for the following scenarios: normal operation, data flooding attack at server, proposed security system and proposed defense response to data flooding attack at server

Runs	Successfully Delivered Packet Number (over 900 seconds)	
Service Requester Node	21	30
Normal operation	8078	7869
Data flooding attack at server	2463	1404
Proposed security system	6789	7730
Proposed defense response to data flooding attack at server	2848	5344

The defense system improves the delivery rate of the individual and overall service traffic by eliminating the attacking packets while retransmitting the legitimate

packets. Figure 34, Figure 35 and Table 9 show that the defense system helps to increase legitimate packet delivery. In this case, the delivered packets of node 21 in 900 seconds are improved by 15.63%, and the delivered packets of node 30 are improved by 280.63%. When the DDoS attackers gather around a specific server and directly attack it, the guard nodes can pull the malicious traffic away from the server. When the packets are jammed before they can reach the guard nodes, the improvement on legitimate delivery is not significant, but if the redirection mechanism tuned the traffic and save the bandwidth out of the malicious traffic, the legitimate traffic may gain a tremendous improvement.

5.2.1.4 End-to-end Delay

The individual end-to-end delays of legitimate packets in the following scenarios: of normal operation, data flooding attack at server, proposed security system and proposed defense response to data flooding attack at server are illustrated respectively in Figure 36, and they are illustrated in Figure 37.

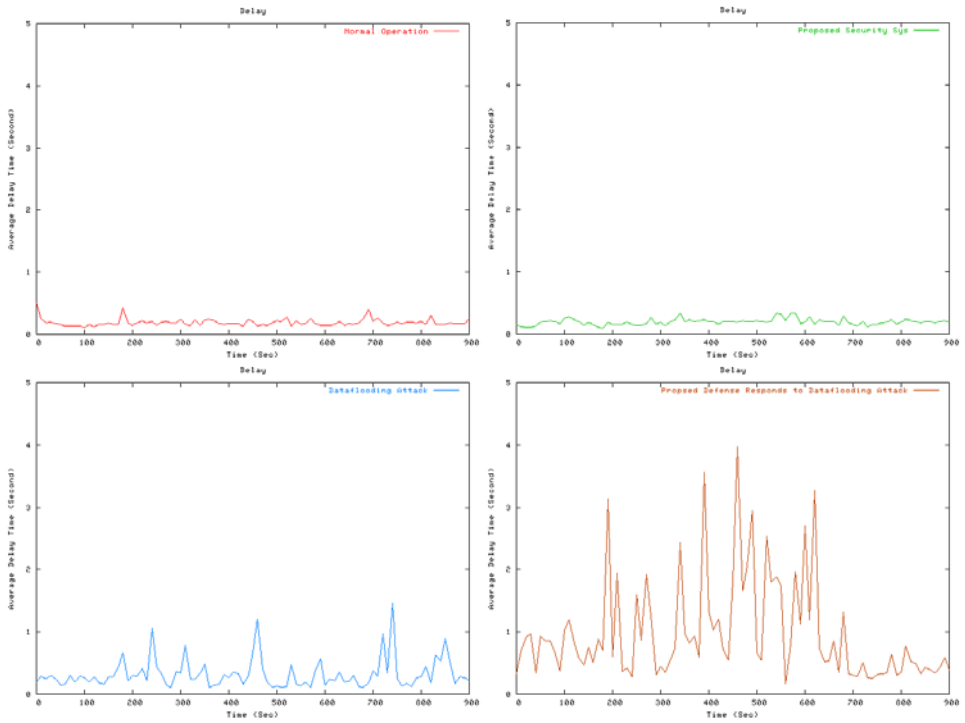


Figure 36. Individual End-to-End Delays in Scenarios Normal Operation, Data Flooding Attack at Server, Proposed Security System and Proposed Defense Response to Data Flooding Attack at Server

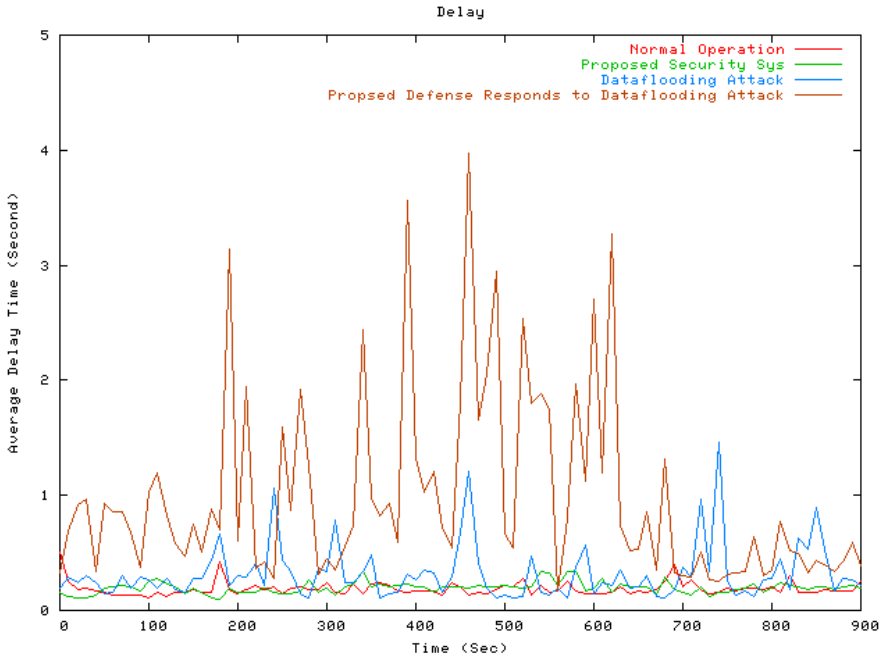


Figure 37. End-to-End Delays in Scenarios Normal Operation, Data Flooding Attack at Server, Proposed Security System and Proposed Defense Response to Data Flooding Attack at Server

Table 10. Average end-to-end delays comparison for the following scenarios: normal operation, data flooding attack at server, proposed security system and proposed defense response to data flooding attack at server

Runs	Average Delay (second)
Normal operation	0.187
Data flooding attack at server	0.322
Proposed security system	0.197
Proposed defense response to data flooding attack at server	0.977

The defense system may take longer processing delay time and retransmission attempts to send some certain packets successfully, but gain higher overall delivery rate. The tradeoff is between longer average end-to-end delay time and a more stable system performance.

5.2.1.5 System Overhead

The defense system overhead is computed by comparing packet numbers of routing, legitimate data and system management packets between the runs without the defense system and with the defense system. The system overhead of the proposed security system over normal operation is illustrated in Figure 38 and the system overhead of the proposed defense response to data flooding attack at server over data flooding attack at server is illustrated in Figure 39.

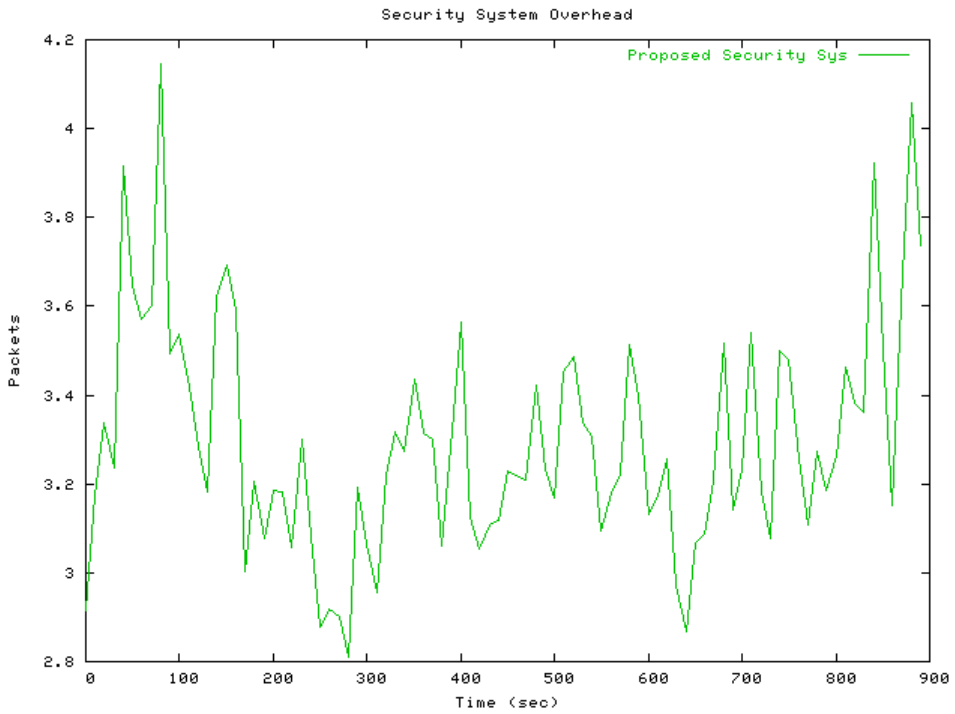


Figure 38. System Overhead of Proposed Security System over Normal Operation

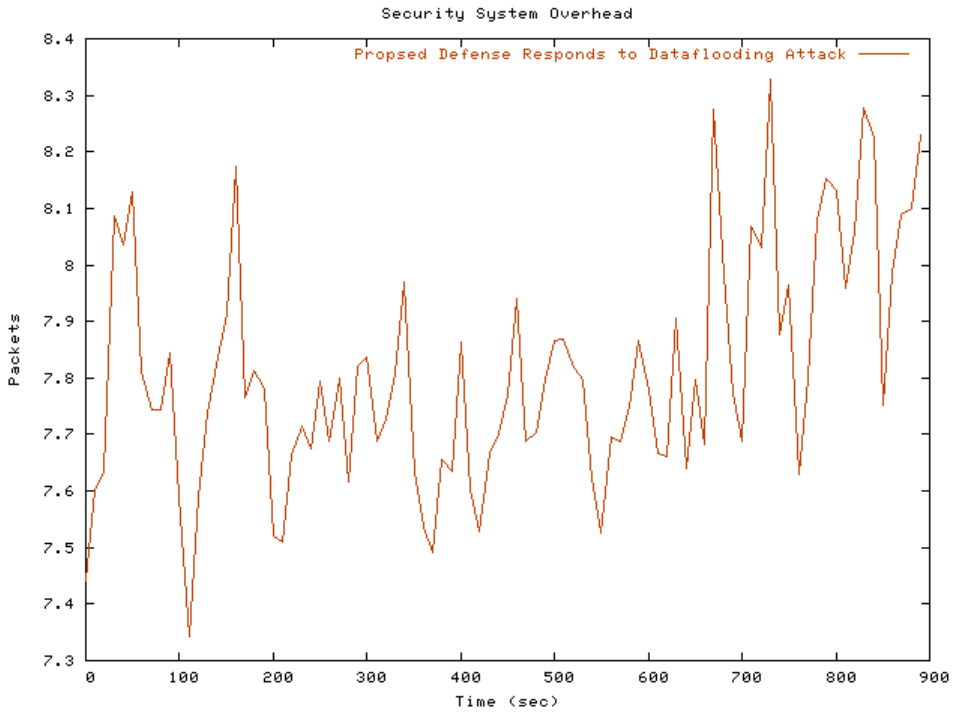


Figure 39. System Overhead of Proposed Defense Response to Data Flooding Attack at Server Over Data Flooding Attack at Server

Table 11. Average system overhead comparison for the following scenarios: proposed security system over normal operation and proposed defense response to data flooding attack at server over data flooding attack at server

Runs	System Overhead (Packets per Second)
Proposed security system over normal operation	3.30
Proposed defense response to data flooding attack at server over data flooding attack at server	7.81

The defense system detours the packets, which takes more packets/hops to finish a same transmission. When the network is under data flooding attacks, the defense system's retransmission attempts increase the network workload.

5.2.2 Results Comparison for the Following Scenarios: Normal Operation, Data Flooding Attack at Requester, Proposed Security System and Proposed Defense Response to Data Flooding Attack at Requester

In this set of experiments, service provider is node 35, which is known only by the defense system nodes. The service is broadcasted with the pseudo service ID of 49 in the proposed security system, which is known by all nodes in the network. The guard nodes of the defense system intercept the traffic to the service ID 49 and forward to service provider node 35. Service requesters are nodes 21 and 30. Five attackers are nodes 28, 18, 6, 43 and 1. During the data flooding attack scenarios, the attackers are deployed nearby requester node 21, and sending the data packets to the server, which results in a traffic congesting and dropping packets around the requester node. Once the run starts, all nodes, include the server node, the proposed defense system nodes, service

requester nodes, and attacker nodes are randomly roaming, which means at an arbitrary moment, a node is moving at a random speed and toward a random direction.

5.2.2.1 Throughput

Table 12. Overall network throughput comparison for the following scenarios: normal operation, data flooding attack at requester node, proposed security system and proposed defense response to data flooding attack at request node over 900 seconds.

Runs	Overall Throughput (MBytes)
Normal operation	7.79
Data flooding attack at requester	2.23
Proposed security system	7.09
Proposed defense response to data flooding attack at requester	4.66

By suppressing the attack traffic, the defense system helps the network gain more useful throughput when DDoS data flooding attacks occurring at one side of the network. Table 12 shows the defense system improves the throughput by 108.97% when the network region near the requester node 21 is under attack.

5.2.2.2 Dropped Packets

The overall packet drop rate of the following scenarios: normal operation, data flooding attack at a requester, proposed security system and proposed defense response to data flooding attack at requester are illustrated in Figure 40. The overall legitimate packet drop rates are illustrated in Figure 41. The overall packet drop rates of four scenarios are displayed in four colored lines. The rates of normal operation (in red) and

proposed security system (in green) are close to 0, but the rates of data flooding attack at server (in blue) and proposed defense response to data flooding attack at server (in brown) are close to 200 packets. The overall attacking packet drop rates are illustrated in Figure 42.

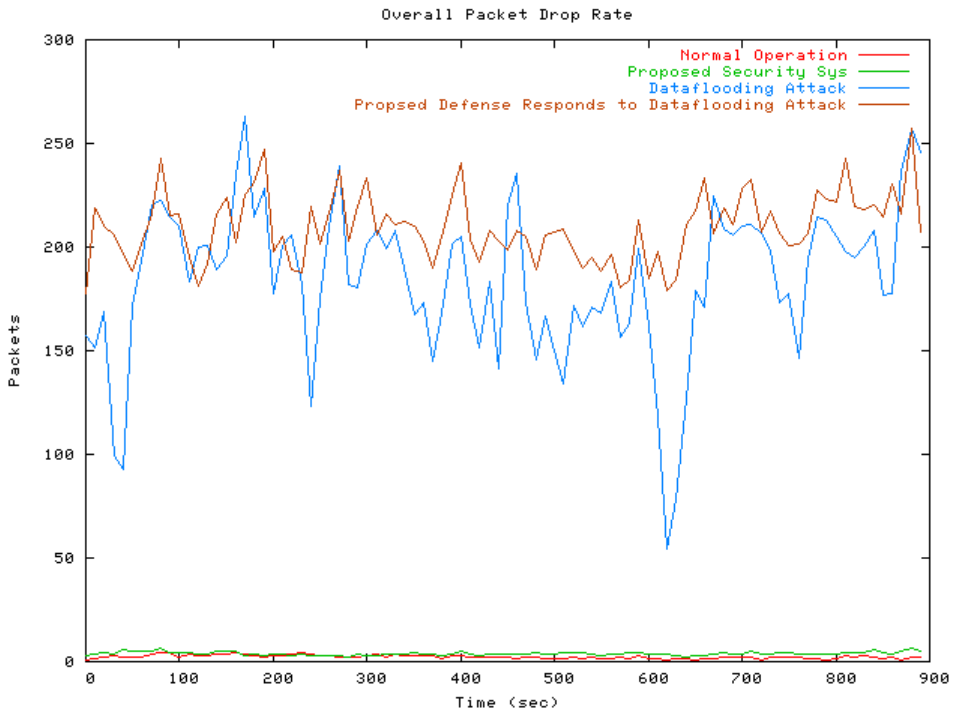


Figure 40. Overall Packet Drop Rate in Scenarios Normal Operation, Data Flooding Attack at Requester, Proposed Security System and Proposed Defense Response to Data Flooding Attack at Requester

Table 13. Overall drop rate comparison for the following scenarios: of normal operation, data flooding attack at requester, proposed security system and proposed defense response to data flooding attack at requester

Runs	Overall drop rate (packets per second)
Normal operation	2.28
Data flooding attack at requester	185.20
Proposed security system	3.87
Proposed defense response to data flooding attack at requester	209.39

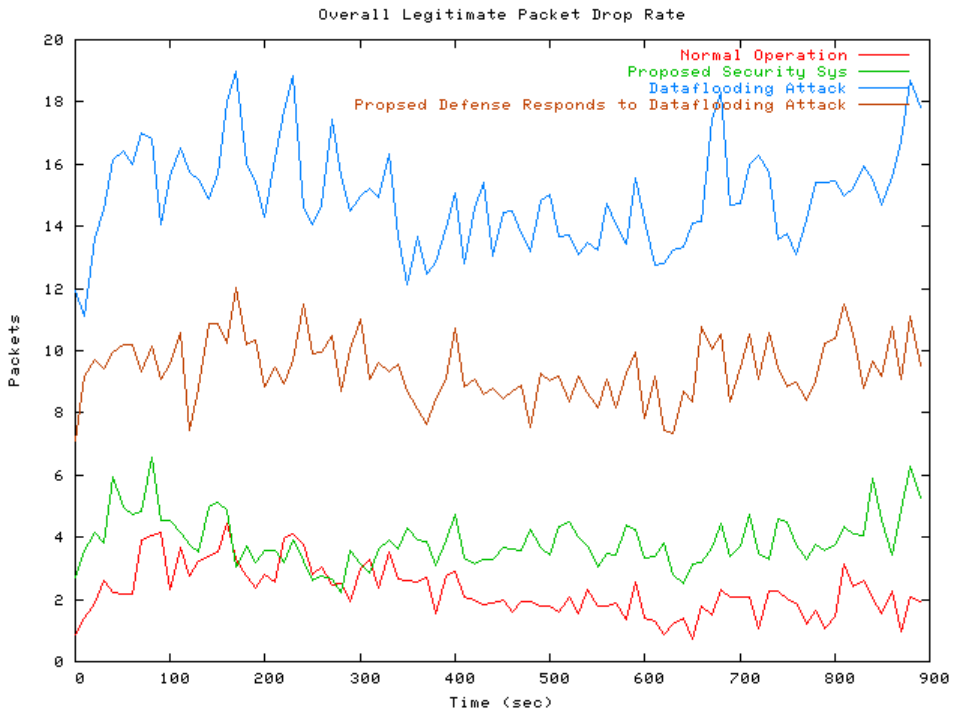


Figure 41. Overall Legitimate Packet Drop Rate in Scenarios Normal Operation, Data Flooding Attack at Requester, Proposed Security System and Proposed Defense Response to Data Flooding Attack at Requester

Table 14. Overall legitimate packet drop rate comparison for the following scenarios: normal operation, data flooding attack at requester, proposed security system and proposed defense response to data flooding attack at requester

Runs	Overall legitimate packet drop rate (packets per second)
Normal operation	2.28
Data flooding attack at requester	14.93
Proposed security system	3.87
Proposed defense response to data flooding attack at requester	9.39

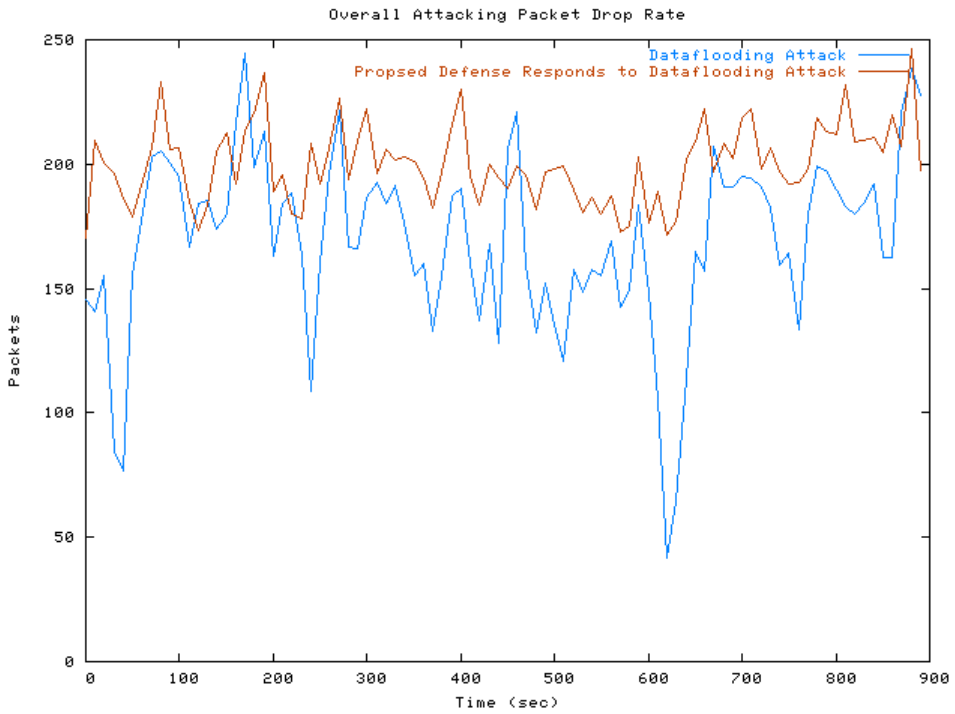


Figure 42. Overall Attacking Packet Drop Rate in Scenarios Data Flooding Attack at Requester and Proposed Defense Response to Data Flooding Attack at Requester

Table 15. Overall attacking packet drop rate comparison for the following scenarios: of normal operation, data flooding attack at requester, proposed security system and proposed defense response to data flooding attack at requester

Runs	Overall attacking packet drop rate (packets per second)
Data flooding attack at requester	170.27
Proposed defense response to data flooding attack at requester	200.00

When one side of the network is under data flooding attack, Figure 40 and Table 13 show the defense system reduces the overall network traffic which is mostly attacking traffic; meanwhile, the defense system drops 37.10% less legitimate packets as shown in Figure 41 and Table 14. Most attacking packets dropped in an unprotected network are

caused by the natural congestion. But the defense system intentionally filters all those attacking packets over the effect of attacking congestion. Figure 42 and Table 15 show there are no attacking packets going through the defense system.

Because of the detour mechanism of the defense system, the part of the network that is under the data flooding attack is relieved.

5.2.2.3 Delivery Rate

The overall delivered packets per second from the service requester node 21 in the following scenarios: normal operation, data flooding attack at requester, proposed security system and proposed defense response to data flooding attack at requester are illustrated in Figure 43. The delivered packets from the service requester node 30 are illustrated in Figure 44. The brown line is overall higher than blue line, which means there are more packets delivered in the network with the defense system than without the defense system.

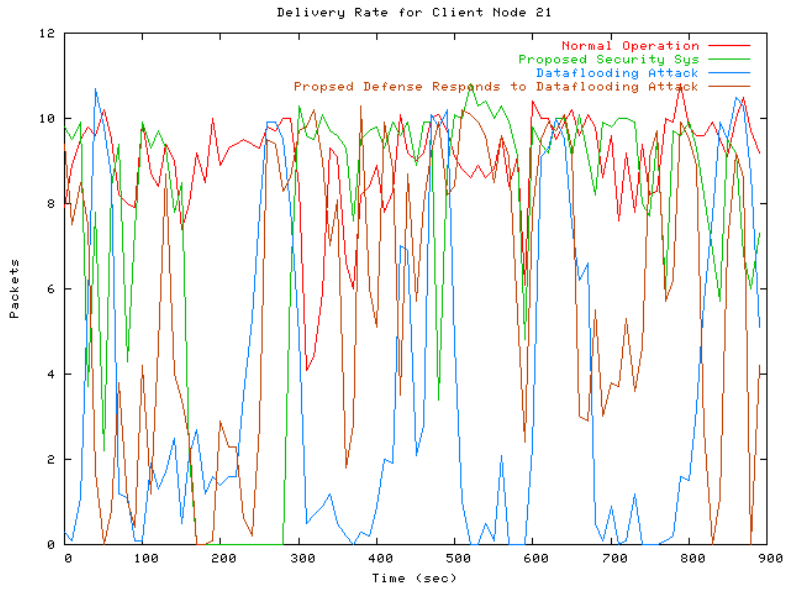


Figure 43. Delivered Packets for Node 21 in Scenarios Normal Operation, Data Flooding Attack at Requester, Proposed Security System and Proposed Defense Response to Data Flooding Attack at Requester

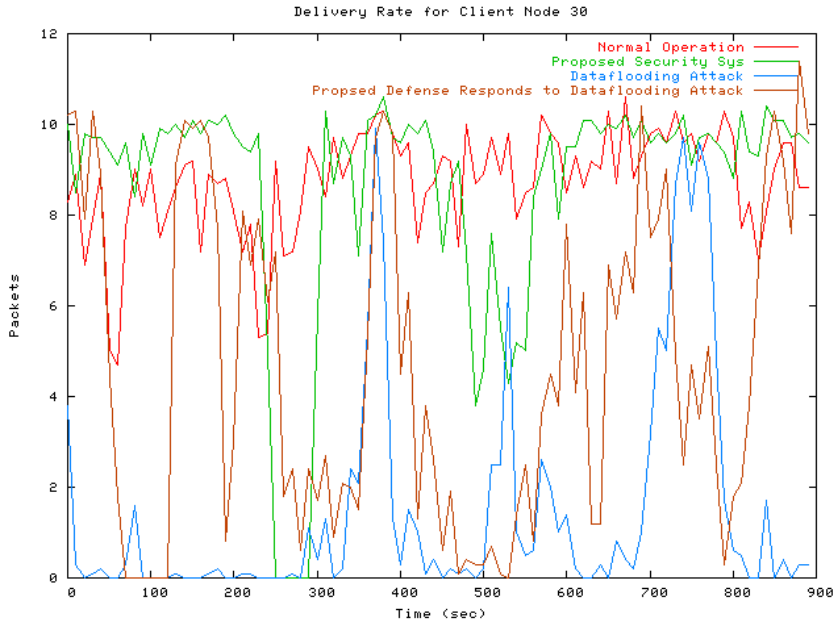


Figure 44. Delivered Packets for Node 30 in Scenarios Normal Operation, Data Flooding Attack at Requester, Proposed Security System and Proposed Defense Response to Data Flooding Attack at Requester

Table 16. Successfully delivered packets over 900 seconds comparison for the following scenarios: normal operation, data flooding attack at requester, proposed security system and proposed defense response to data flooding attack at requester.

Runs	Successfully Delivered Packet Number (over 900 seconds)	
Service Requester Node	21	30
Normal operation	8078	7869
Data flooding attack at requester	3216	1350
Proposed security system	6789	7730
Proposed defense response to data flooding attack at requester	5347	4202

The defense system improves the delivery rate of the individual and overall service traffic by eliminating the attacking packets while retransmitting the legitimate packets. Figure 43, Figure 44 and Table 16 show the defense system helps to increase the legitimate delivery. In this case, the delivered packets of node 21 are improved by 66.26%, and the delivered packets of node 30 are improved by 211.26%. When data flooding attackers gather at one side of the network, which in this case is close to the requester node 21, the nearby local guard pulls the most attack traffic, so that the rest of the network gains higher throughput. At the same time, the defense system helps the victim requester forward the packets out of range of attackers.

5.2.2.4 End-to-end Delay

The individual end-to-end delay of the legitimate packets in the following scenarios: normal operation, data flooding attack at requester, proposed security system

and proposed defense response to data flooding attack at requester are illustrated respectively in Figure 45, and they are illustrated in Figure 46.

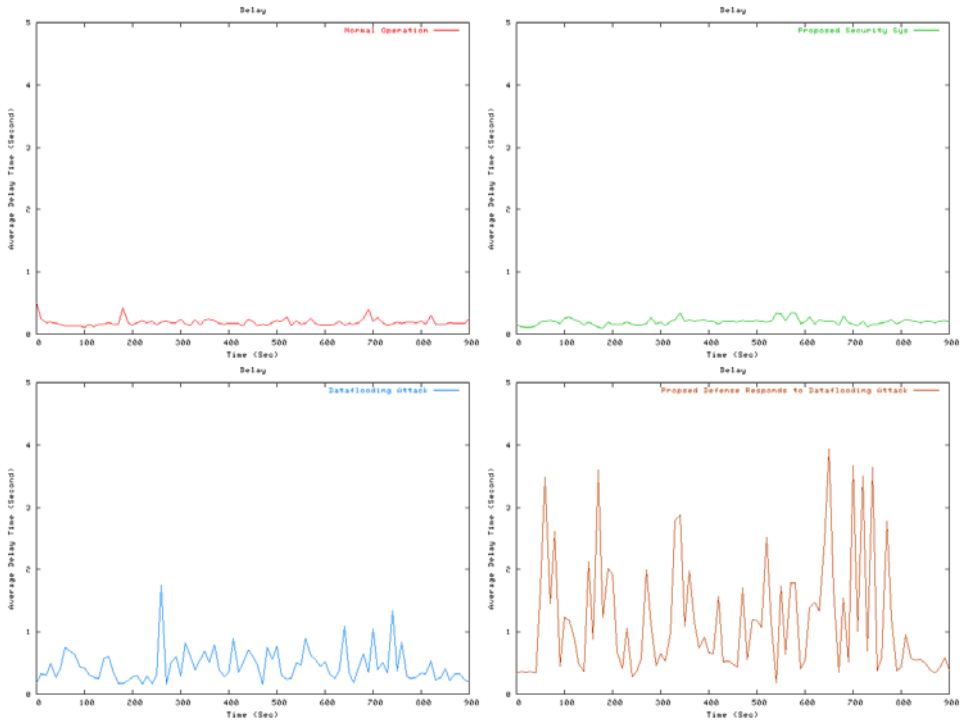


Figure 45. Individual End-to-End Delays in Scenarios Normal Operation, Data Flooding Attack at Requester, Proposed Security System and Proposed Defense Response to Data Flooding Attack at Requester

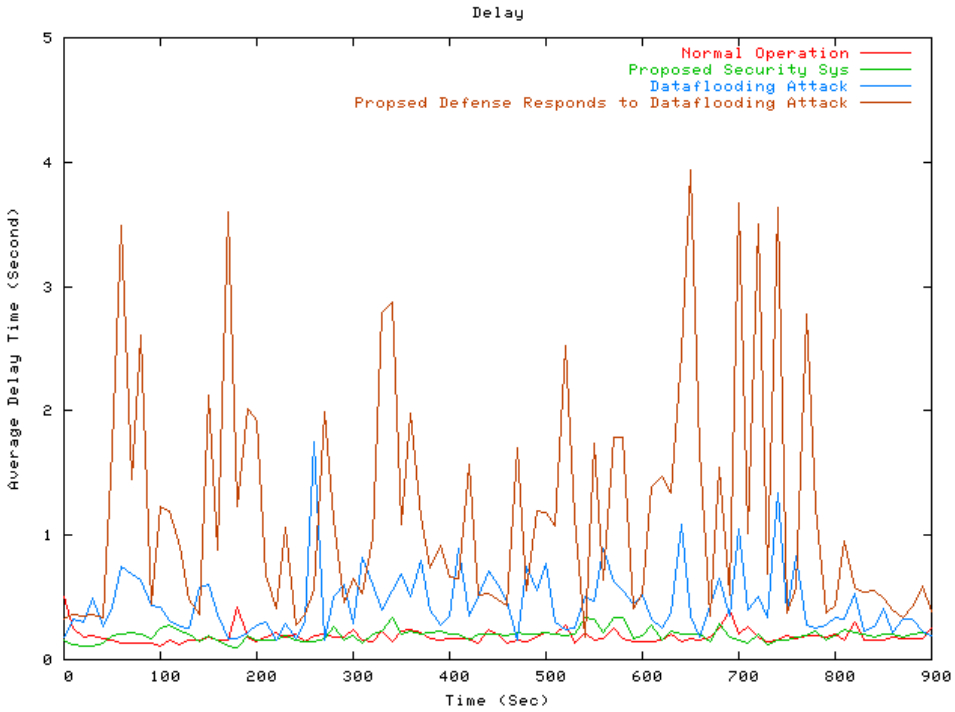


Figure 46. End-to-End Delays in Scenarios Normal Operation, Data Flooding Attack at Requester, Proposed Security System and Proposed Defense Response to Data Flooding Attack at Requester

Table 17. Average end-to-end delays comparison for the following scenarios: normal operation, data flooding attack at requester, proposed security system and proposed defense response to data flooding attack at requester

Runs	Delay (second)
Normal operation	0.187
Data flooding attack at requester	0.459
Proposed security system	0.197
Proposed defense response to data flooding attack at requester	1.183

The defense system may take longer processing delay time and retransmission attempts to send some certain packets successfully, but gains higher overall delivery rate. The tradeoff is between longer average end-to-end delay time and a more stable system performance.

5.2.2.5 System Overhead

The defense system overhead is computed by comparing packet numbers of routing, legitimate data and system management packets between the runs without the defense system and with the defense system. The system overhead of the proposed security system over normal operation is illustrated in Figure 38 and the system overhead of proposed defense response to data flooding attack at requester over data flooding attack at requester is illustrated in Figure 47.

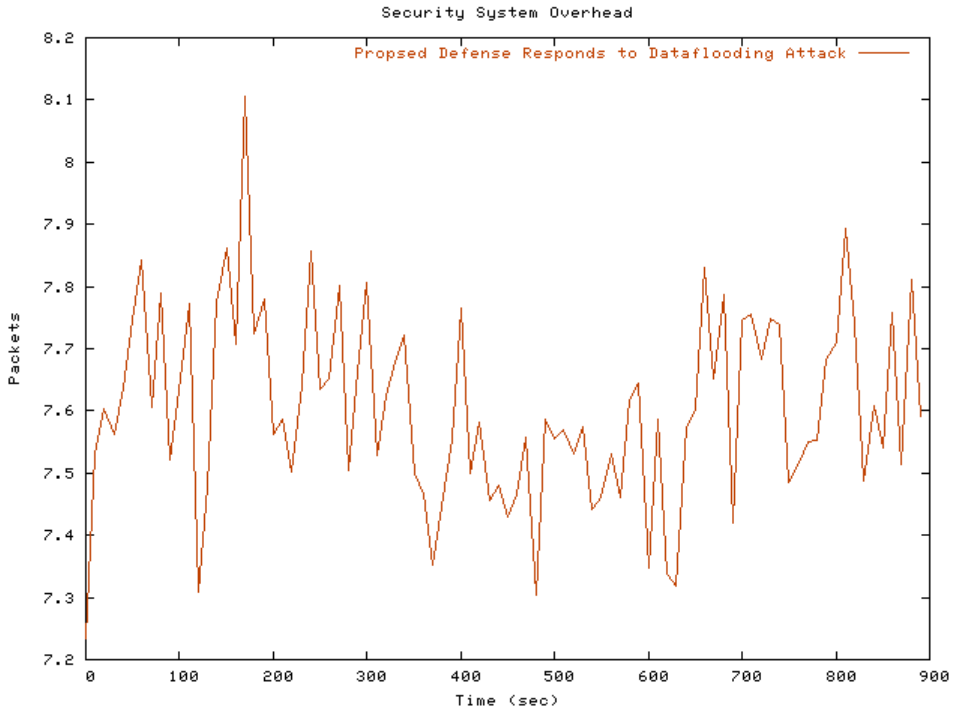


Figure 47. System Overhead of Proposed Defense Response to Data Flooding Attack at Server Over Data Flooding Attack at Requester

Table 18. Average system overhead comparison for the following scenarios: proposed security system over normal operation and proposed defense response to data flooding attack at requester over data flooding attack at requester

Runs	System Overhead (Packets per Second)
Proposed security system over normal operation	3.30
Proposed defense response to data flooding attack at requester over data flooding attack at requester	7.60

The defense system detours the packets, which takes more packets/hops to finish a same transmission. When the network is under data flooding attacks, the defense system's retransmission attempts increase the network workload.

5.2.3 Results Comparison for the Following Scenarios: Normal Operation, Random Data Flooding Attack, Proposed Security System and Proposed Defense Response to Random Data Flooding Attack

In this set of experiments, service provider is node 35, which is known only by the defense system nodes. The service is broadcasted with the pseudo service ID of 49 in the proposed security system, which is known by all nodes in the network. The guard nodes of the defense system intercept the traffic to the service ID 49 and forward to service provider node 35. Service requesters are nodes 21 and 30. Five attackers are nodes 22, 28, 0, 33 and 18. At the start point of the scenarios of random data flooding attack and proposed defense response to random data flooding attack, the attacker group is deployed randomly in the network, and sending the data packets to the server. Once the run starts, all nodes, include the server node, the proposed defense system nodes,

service requester nodes, and attacker nodes are randomly roaming, which means at an arbitrary moment, a node is moving at a random speed and toward a random direction.

5.2.3.1 Throughput

Table 19. Overall network throughput comparison for the following scenarios: normal operation, random data flooding attack, proposed security system and proposed defense response to random data flooding attack over 900 seconds.

Runs	Overall Throughput (MBytes)
Normal operation	7.79
Random data flooding attack	2.43
Proposed security system	7.09
Proposed defense response to random data flooding attack	4.71

By suppressing the attack traffic, the defense system helps the network gain more useful throughput under a regular random DDoS data flooding attacks. Table 19 shows the defense system improves the overall throughput by 93.83% when the network is under a regular random attack. The legitimate packets dropped before reaching the guard nodes are not rescued by the defense system.

5.2.3.2 Dropped Packets

The overall packet drop rate of the following scenarios: normal operation, random data flooding attack, proposed security system and proposed defense response to random data flooding attack are illustrated in Figure 48. The overall legitimate packet drop rates are illustrated in Figure 49. The overall packet drop rates of four scenarios are displayed

in four colored lines. The drop rates of normal operation (in red) and the proposed security system (in green) are close to 0, but the drop rates of data flooding attack at server (in blue) and proposed defense response to data flooding attack at server (in brown) are close to 200 packets. The overall attacking packet drop rates are illustrated in Figure 50.

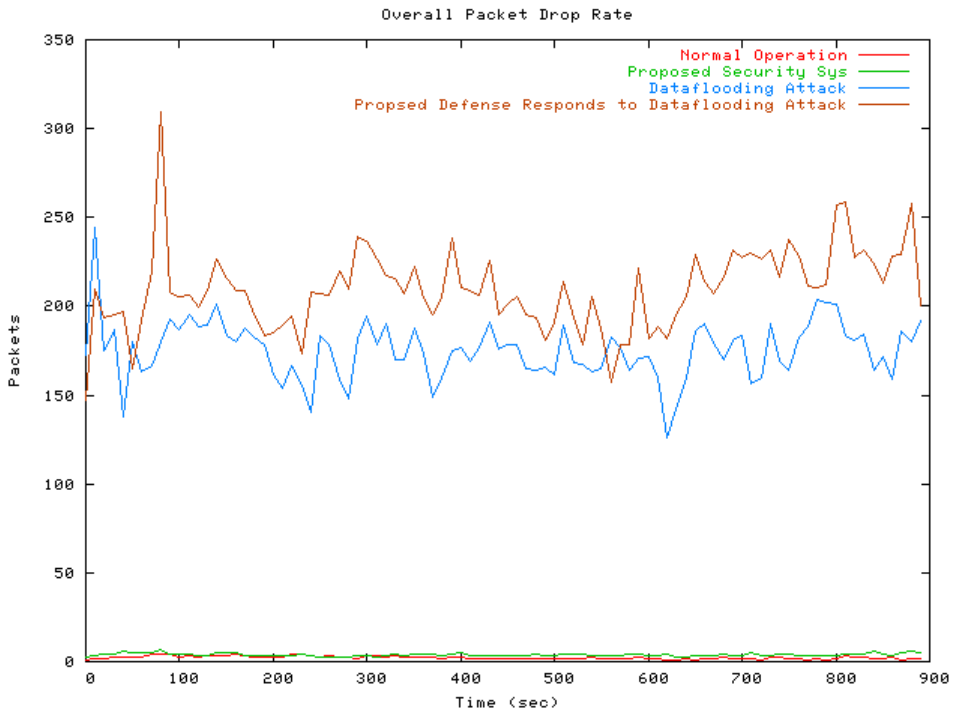


Figure 48. Overall Packet Drop Rate in Scenarios Normal Operation, Random Data Flooding Attack, Proposed Security System and Proposed Defense Response to Random Data Flooding Attack

Table 20. Overall drop rate comparison for the following scenarios: normal operation, random data flooding attack, proposed security system and proposed defense response to random data flooding attack

Runs	Overall drop rate (packets per second)
Normal operation	2.28
Random data flooding attack	175.16
Proposed security system	3.87
Proposed defense response to random data flooding attack	209.28

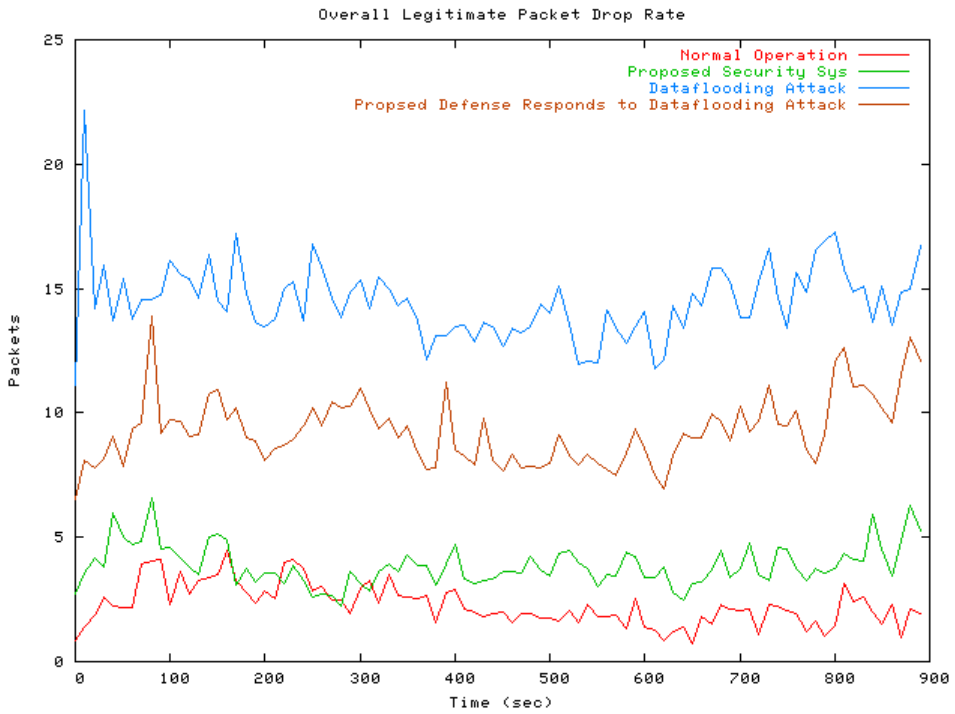


Figure 49. Overall Legitimate Packet Drop Rate in Scenarios Normal Operation, Random Data Flooding Attack, Proposed Security System and Proposed Defense Response to Random Data Flooding Attack

Table 21. Overall legitimate packet drop rate comparison for the following scenarios: normal operation, random data flooding attack, proposed security system and proposed defense response to random data flooding attack

Runs	Overall legitimate packet drop rate (packets per second)
Normal operation	2.28
Random data flooding attack	14.46
Proposed security system	3.87
Proposed defense response to random data flooding attack	9.28

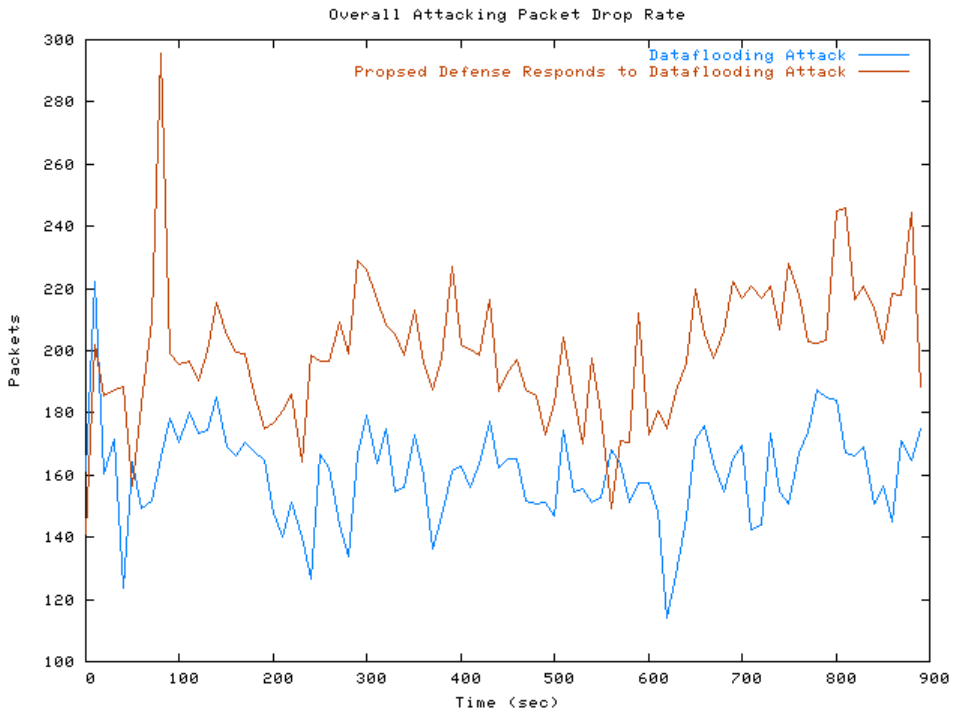


Figure 50. Overall Attacking Packet Drop Rate in Scenarios Random Data Flooding Attack and Proposed Defense Response to Random Data Flooding Attack

Table 22. Overall attacking packet drop rate comparison for the following scenarios: normal operation, random data flooding attack, proposed security system and proposed defense response to random data flooding attack

Runs	Overall attacking packet drop rate (packets per second)
Random data flooding attack	160.70
Proposed defense response to random data flooding attack	200.00

When the network is under a random data flooding attack, Figure 48 and Table 20 show the defense system reduces the network traffic which is mostly attack traffic; meanwhile, the defense system drops 35.82% less legitimate packets as shown in Figure 49 and Table 21. Most attacking packets dropped in an unprotected network are caused by naturally occurring congestion. But the defense system intentionally filters all those

attacking packets over the effect of attacking congestion. Figure 50 and Table 22 show there are no attacking packets going through the defense system.

5.2.3.3 Delivery Rate

The overall delivered packets per second from the service requester node 21 in the following scenarios: normal operation, random data flooding attack, proposed security system and proposed defense response to random data flooding attack are illustrated in Figure 51. The delivered packets from the service requester node 30 are illustrated in Figure 52. The brown line is overall higher than blue line, which means there are more packets delivered in the network with the defense system than without the defense system.

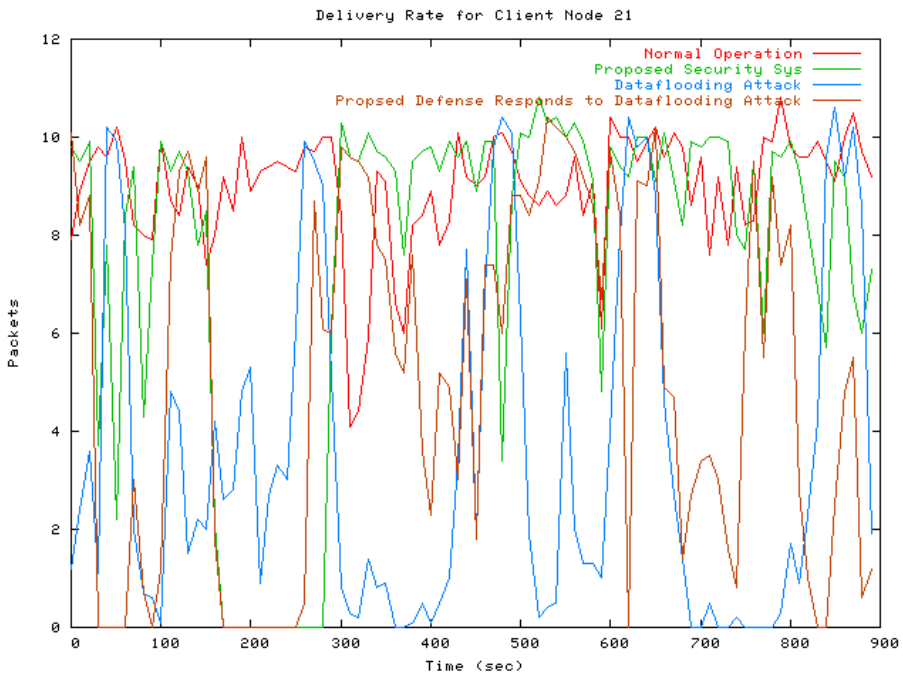


Figure 51. Delivered Packets for Node 21 in Scenarios Normal Operation, Random Data Flooding Attack, Proposed Security System and Proposed Defense Response to Random Data Flooding Attack

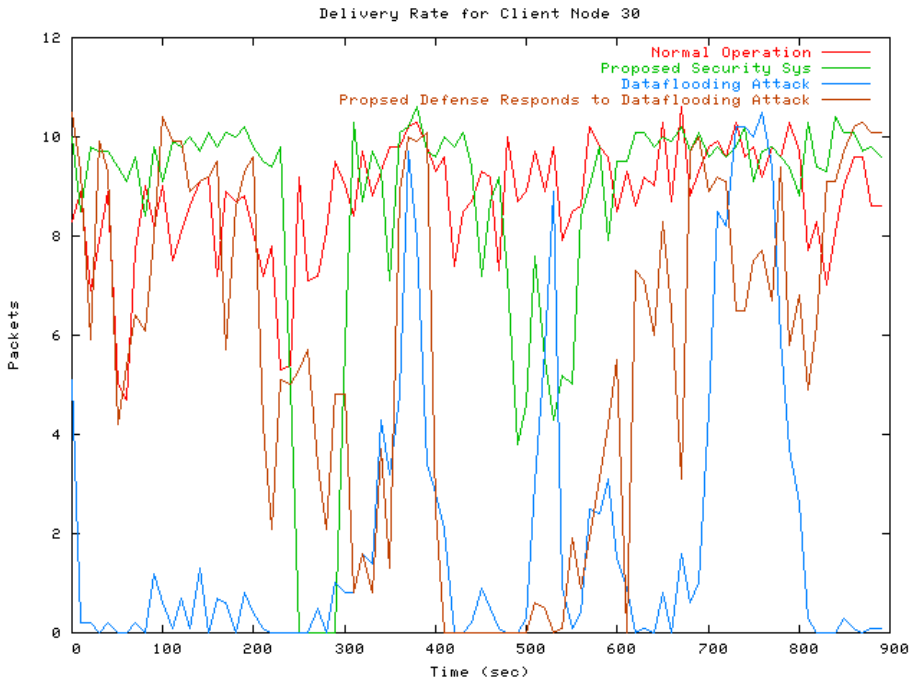


Figure 52. Delivered Packets for Node 30 in Scenarios Normal Operation, Random Data Flooding Attack at Requester, Proposed Security System and Proposed Defense Response to Random Data Flooding Attack

Table 23. Successfully delivered packets over 900 seconds comparison for the following scenarios: normal operation, random data flooding attack, proposed security system and proposed defense response to random data flooding attack

Runs	Successfully Delivered Packet Number (over 900 seconds)	
Service Requester Node	21	30
Normal operation	8078	7869
Random data flooding attack	3212	1769
Proposed security system	6789	7730
Proposed defense response to random data flooding attack	4572	5077

The defense system improves the delivery rate of the individual and overall service traffic by eliminating the attacking packets while retransmitting the legitimate packets. Figure 51, Figure 52 and Table 23 show that the defense system helps to increase the legitimate delivery. In this case, the delivered packets of node 21 in 900 seconds are improved by 42.34%, and the delivered packets of node 30 are improved by 187.00%. When the DDoS attackers gather around a specific server and directly attack it, the guard nodes can pull the malicious traffic away from the server. When the packets are dropped because of the congestion before they can reach the guard nodes, the improvement on legitimate delivery is not significant, but if the redirection mechanism tuned the traffic and save the bandwidth out of the malicious traffic, the legitimate traffic may gain a tremendous improvement.

5.2.3.4 End-to-end Delay

The individual end-to-end delays of the legitimate packets in the following scenarios: normal operation, random data flooding attack, proposed security system and proposed defense response to random data flooding attack are illustrated respectively in Figure 53, and they are illustrated in Figure 54.

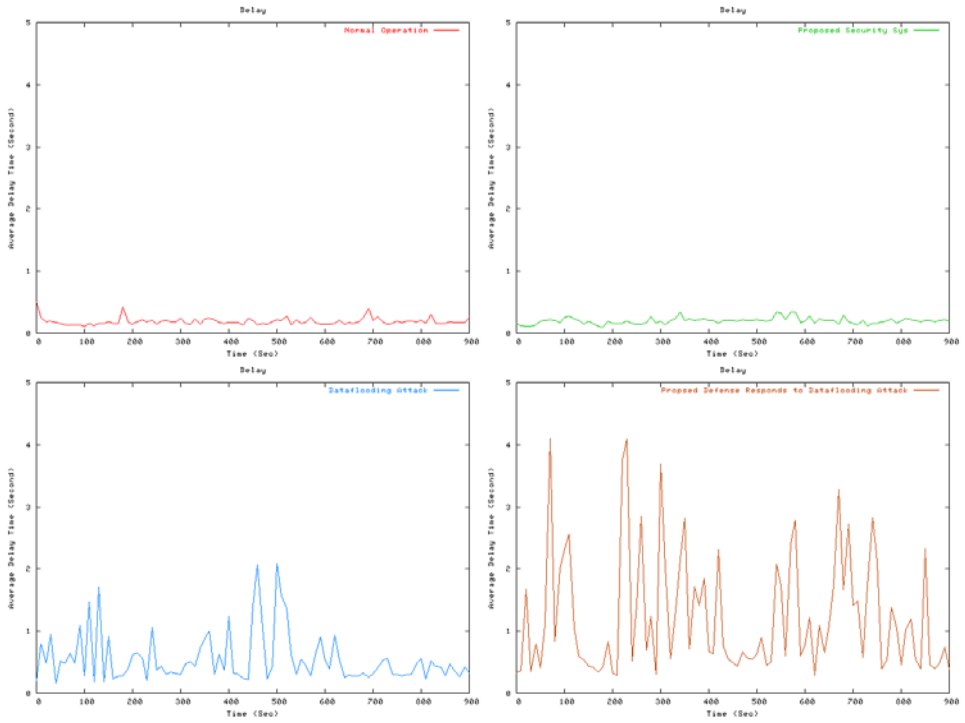


Figure 53. Individual End-to-End Delays in Scenarios Normal Operation, Random Data Flooding Attack, Proposed Security System and Proposed Defense Response to Random Data Flooding Attack

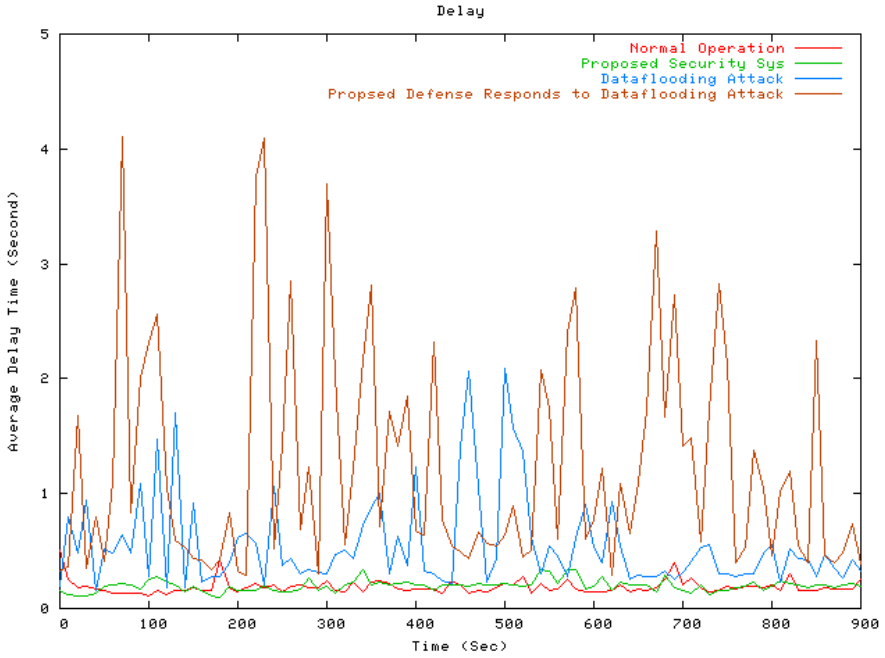


Figure 54. End-to-End Delays in Scenarios Normal Operation, Random Data Flooding Attack, Proposed Security System and Proposed Defense Response to Random Data Flooding Attack

Table 24. Average end-to-end delays comparison for the following scenarios: normal operation, random data flooding attack, proposed security system and proposed defense response to random data flooding attack

Runs	Average Delay (second)
Normal operation	0.187
Random data flooding attack	0.556
Proposed security system	0.197
Proposed defense response to random data flooding attack	1.230

The defense system may take longer processing delay time and retransmission attempts to send some certain packets successfully, but gain higher overall delivery rate. The tradeoff is between longer average end-to-end delay time and a more stable system performance.

5.2.3.5 System Overhead

The defense system overhead is computed by comparing packet numbers of routing, legitimate data and system management packets between the runs without the defense system and with the defense system. The system overhead of the proposed security system over normal operation is illustrated in Figure 38 and the system overhead of the proposed defense response to random data flooding attack is illustrated in Figure 55.

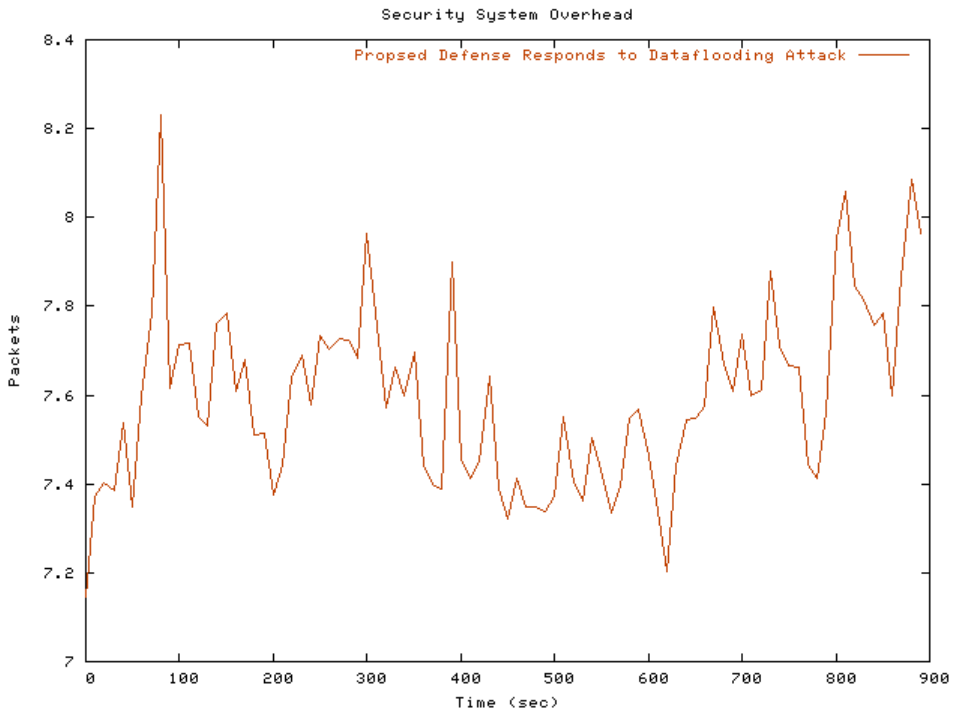


Figure 55. System Overhead of Proposed Defense Response to Random Data Flooding attack

Table 25. Average system overhead comparison for the following scenarios: proposed security system over normal operation and proposed defense response to random data flooding attack

Runs	System Overhead (Packets per Second)
Proposed security system over normal operation	3.30
Proposed defense response to random data flooding attack	7.60

The defense system detours the packets, which takes more packets/hops to finish a same transmission. When the network is under data flooding attacks, the defense system's retransmission attempts increase the network workload.

5.2.4 Summary of Defense System Response to Data Flooding Attacks

The three sets of experiments are designed to test the worst situation for the server, the worst situation for the requester and a random attack. With sufficient run time and fully random mobility, the experiment results are close and show the common performance of the proposed defense system. When the normal operation network is under DDoS data flooding attacks, the network resources and functionalities are tremendously impacted and damaged. The proposed defense system may effectively detour and filter the network traffic at the guard nodes. At the same time, the defense system reinforces the legitimate traffic to fight against the attacking traffic.

5.3 Defense System on Black-hole Attacks

The experiment results of the following scenarios: Black-hole Attack 1, Proposed Defense System Response to Black-hole Attack 1, Black-hole Attack 2 and Proposed Defense System Response to Black-hole Attack 2 show that the nodes being close to the black-hole attacker are deceived by the attacker, and their routes to S are directed to the attacker. But the attacker cannot feign itself as one of the guard nodes. It takes only one ordinary RREQ for the defense system to reveal the attacker node.

5.4 Experiment Results Summary

The defense system is implemented and evaluated against three types of DoS/DDoS attack. The experiment results reveal the tolerance aspect of the AODV routing protocol against RREQ flooding attack. Three sets of experiments show the proposed defense system may protect the network resource and improve the network

functionalities when the network is under DDoS data flooding attacks (Table 26, Table 27 and Table 28). Depends on attacking location and direction in the network, the impact on legitimate nodes is different. Because the defense system can detour all service traffic toward the guard nodes and drop all the malicious traffic at the guard nodes, the defense system reduces the network traffic and cleans up the network region between the guard nodes and the service provider. The defense system may improve a service requester delivery rate tremendously if the malicious traffic is perfectly suppressed. The defense system may help very little if the malicious traffic still drops most requester packets. Because the overhead introduced by the defense system is much less than the attacking traffic, the performance decreased by the system overhead is very slight, it is possible the defense system may decrease the performance at some extreme situations, such as a service requester and a service provide are side by side, but all guard nodes are far away. The proposed defense system can discover any black-hole attacker in the network.

Table 26. System Performance Summery for Data Flooding Attack at Server

	Runs		Difference
	Data Flooding Attack at Server	Proposed Defense Response to Data Flooding Attack at Server	
Throughput (Mbytes in 900sec)	1.89	4.00	111.64%
Overall Dropped Packets (Packets/sec)	198.55	210.96	6.25%
Overall Dropped Legitimate Packets (Packets/sec)	15.70	10.90	-30.57%
Overall Dropped Attacking Packets (Packets/sec)	182.85	200.00	All Dropped
Delivered Packets for Requester Node 21 (in 900sec)	2463	2848	15.63%
Delivered Packets for Requester Node 30 (in 900sec)	1404	5344	280.63%
End to End Delay (sec)	0.322	0.977	203.42%

Table 27. System Performance Summery for Data Flooding Attack at Requester

	Runs		Difference
	Data Flooding Attack at Requester	Proposed Defense Response to Data Flooding Attack at Requester	
Throughput (Mbytes in 900sec)	2.23	4.66	108.97%
Overall Dropped Packets (Packets/sec)	185.20	209.39	13.06%
Overall Dropped Legitimate Packets (Packets/sec)	14.93	9.39	-37.10%
Overall Dropped Attacking Packets (Packets/sec)	170.27	200.00	All Dropped
Delivered Packets for Requester Node 21 (in 900sec)	3216	5347	66.26%
Delivered Packets for Requester Node 30 (in 900sec)	1350	4202	211.26%
End to End Delay (sec)	0.459	1.183	157.73%

Table 28. System Performance Summery for Random Data Flooding Attack

	Runs		Difference
	Random Data Flooding Attack	Proposed Defense Response to Random Data Flooding Attack	
Throughput (Mbytes in 900sec)	2.43	4.71	93.83%
Overall Dropped Packets (Packets/sec)	175.16	209.28	19.48%
Overall Dropped Legitimate Packets (Packets/sec)	14.46	9.28	-35.82%
Overall Dropped Attacking Packets (Packets/sec)	160.70	200.00	All Dropped
Delivered Packets for Requester Node 21 (in 900sec)	3212	4572	42.34%
Delivered Packets for Requester Node 30 (in 900sec)	1769	5077	187.00%
End to End Delay (sec)	0.556	1.230	121.22%

CHAPTER 6

EXPERIMENT DISCUSSION

6.1 Advantages of the Proposed Defense System

The defense system works without assistance from other nodes. No system-wide upgrade or modification is needed. The defense system nodes are hosted on the same regular network device as other common network nodes. The defense nodes comply with the same network protocol interface as the other standard network nodes do. Standard AODV packets are used by the routing packets and data packets between two system nodes or a system node and a network node. The defense system in the experiment involves only four assistant guard nodes, but the network-wide service oriented traffic is efficiently redirected and filtered. All defense methods perform only on the system nodes, but they can redirect the attack traffic in the network and stop it at the guard nodes. The system nodes have good anonymity by regularly applying IP hopping (Appendix C), dual IP and encrypted tunneling.

By redirecting and filtering the service traffic, the network gains a capacity to intercept the flooding attacks from reaching the victim, and limit the damage to a section of the network. If the network does not have this defense system, the flood traffic can travel all the way to attack the victim; but if the network has the defense system, the attack flood will stop at the guard nodes, and the rest of the network can continue to

operate. The different network performance with and without the defense system is illustrated and compared in Chapter 5.

6.2 Redirection of Traffic

Because only the guard nodes respond to the RREQ of the server and announce the route, all subsequent data packets are redirected to the guard nodes according to the AODV protocol. The guard nodes gain full control of the service related traffic. A guard node may decide to drop a packet or forward it according to the security system user's definition. A security system user may decide different filter algorithms to distinguish what traffic is prior to reinforce, what traffic is sacrificable, while what traffic is malicious. The algorithms are based on different network, service application types, user types, service requester types or time. The application-specified filter algorithms are independent to the network-layer security system and will not be discussed in this research.

When one guard node crashes, the service requester routed by the guard node needs to start another route discovery for the service provider, and the request will be responded by another available remote guard node. Then the route is detoured and rebuilt.

6.3 Filtering the Traffic

When a network is under DDoS flooding attacks, and the filter function on the guard nodes starts to work, no attack traffic can go beyond the guard nodes. The

attacking traffic of the network is reduced. When a network section is damaged by the attack traffic, the guard node of the section can reinforce the received legitimate traffic by retransmitting the packets to the service provider. Then the legitimate packet delivery rates are increased. The guard nodes also control the ingress of the legitimate traffic when the network is under flooding attacks. The guard nodes decide which service traffic is legitimate and kept in service, which has lower priority and will be refused along with the attacking traffic according to the user's definition. A simple but realistic algorithm is used in the experiment that only requesters accepted before the DDoS attacks occurring will be kept as legitimate, all the other requests will be dropped along with the malicious traffic. All of those detection, filtering and reinforcement algorithms are independent to the security system. They are modularized on the guard nodes and can be replaced and updated without impact of the network or the security system.

6.4 Absorbing the Attack Force

Evenly deployed guard nodes will draw the service related traffic to the different direction than directly to the provider. All the service related traffic including the attacking traffic reach the closest guard node first. Then the attacking traffic is localized into the limited region that is responded by the involved guard. Then other parts of the network survive. If the service provider is out of the attacked network region, the service will appear normal to the other unimpaired part of the network. If the service provider is inside the attacked section, the impact of the attack depends on the relative position of the provider and the attack traffic.

6.5 System Overhead

The defense system introduces a small amount of overhead to the network. Experiment results show the average increased traffic (overhead) of 16.50% in normal operation and 38.35% in DDoS data flooding attack. At the network initialization stage, the guard nodes discover the routes to the service provider. Guard nodes use dual IP and IP hopping mechanisms, which require more IP resource. When the network is under DDoS flooding attack, the guard nodes attempt to retransmit the legitimate packets to increase the legitimate data delivery rate. The system overhead is either periodical maintenance packets or retransmitted data packets.

6.6 DDoS RREQ Flooding Attacks

The new draft of AODV (RFC3561, 2003) [170] has a mechanism to mitigate RREQ flooding attacks. RFC3561 defines RREQ_RATELIMIT at 10 request packets per second to constrain the RREQ packet number a node can generate or forward in one second. DDoS RREQ flooding attacks may use up the RREQ_RATELIMIT defined on each network node, which makes the whole network refuse any further RREQ requests until the next 1 second interval. But there is no limit on data packets a node can send or forward in AODV networks. So the routing request limit does not prevent data packet flooding.

Even though RREQ flooding attacks cannot take the entire bandwidth all the time, they can impair the normal routing functionality of networks. Therefore no node in the network can send or refresh routing information when the network is under RREQ flooding attacks. Since the defense system nodes are not special, and their

communication depends on the regular intermediate nodes, their routing requests are vulnerable to RREQ flooding attacks too. If a highly mobile network is under RREQ flooding attack, when a guard node loses the route to the service provider because of a topology change, the guard node is not able to rebuild the route. The defense system fails on the routing discovery as well as the whole network.

6.7 Modification on the Simulator and the Validation

Implementing the proposed defense system does not change the AODV protocol on the standard network nodes. The network behavior and character of these nodes are exactly the same as a standard AODV network. We hacked the AODV protocol by removing the RREQ_RATELIMIT condition check in the code to have an unlimited RREQ transmission for the attack. A user-defined filter function is attached to the receiving module in the guard nodes to decide to tunnel a packet or drop it instead of unconditionally forwarding all the packets. In the experiment, the filter only allows the legitimate traffic. A practical filter in real world will be designed according to the network environment, service type, network user types, commercial model of the network, etc. Any filtering algorithm is an independent module to the defense system. The processing delay of a packet added by a guard node is simulated by the interval between recv() and send(), which is the machine processing time of this packet. All the modifications and extensions of the network program source files are in Appendix A.1.

The system is validated through these methods:

- The simulation results are illustrated and examined with the nam animations (see Appendix B.1).

- The system passed the extreme condition tests and a generic condition test described in Chapter 5. The experiment tests a set of extreme conditions, such as the common network without the defense system or DDoS attacks, the system with the defense system, the DDoS attacks on the common network. One defense scenario is that, in a rectangle simulation topology, two service requester nodes are put in the diagonal corners. One is besieged by the attackers and under the direct attack, while another is at the farthest corner from the attackers.
- In Chapter 5, the system results were collected and analyzed with operational graphics. The results show that the defense system increases the system overall throughput, drops whole attacking traffic, and increases the legitimate packet delivery rates.
- The delivery of different type of packets and the behavior of different system nodes are traced, and they meet the model definitions (see Appendix B.2).

6.8 Summary

The proposed defense system mitigates the DDoS flooding attacks on MANETs. The experiment results in Chapter 5 present the guard nodes of the system redirect and filter particular communication traffic. On the other hand, the guard nodes reinforce the legitimate traffic by retransmitting the failed legitimate packets.

The design and implementation of the defense system was validated and verified by various methods, such as animation tools, extreme condition tests and analysis on

delivery packets and behaviors. These validation and verification show that system implementation is accurate and capable.

CHAPTER 7

SUMMARY AND FUTURE WORK

7.1 Design Summary

This dissertation is the first investigation of a feasible independent overlay-type defense system for MANET. Being much distinct to current security research directions that modify or patch routing protocols, this dissertation proposes a novel defense mechanism without modifying the existing AODV routing protocol or requiring cooperation from any non-system node.

The guard nodes block all attacking packets and reinforce legitimate packets. Therefore, the proposed system significantly mitigates DDoS flooding attacks on the service and the network. The defense system improves the network throughput and the service packet delivery rates, reduces the service packet drop rate, and blocks all attacking packets.

The experiment implementation also shows it is very easy to modify the AODV protocol and deploy the DDoS attack code, with resulting tremendous damage to the network. For example in the AODV-UU implementation, the RREQ_RATELIMIT is ensured by one condition check in the sending module. The attacker program need only disable this condition check to have unlimited RREQ transmission (Appendix A.3). AODV does not have enough security protection against DDoS attacks because AODV is designed for low overhead maximum performance. Even an up-to-date secure network

may face DDoS attacks armed with newer technologies in the future, such as attacks based on newly discovered system flaws. But the new security patches can be very hard to apply to a commercial operating MANET. Some nodes are hard to reach, and some nodes belong to unfriendly owners. This research and dissertation proposes a new security solution to AODV networks or other similar MANET networks, without requiring system-wide updates or modification.

The proposed defense system introduces several guard nodes as proxies to a network service. The service provider is transparent to the outside of the defense system because the guard nodes intercept and tunnel the service stream between the service requesters and provider, when the whole network knows only a pseudo ID for the provider. All the service requests are received and forwarded by the guard nodes. Therefore, the defense system may monitor and control the service traffic, and apply the security strategies inside the system. The experiment results demonstrate the defense system can mitigate DDoS attacks without the cooperation from the non-system nodes. When the network is under a data flooding attack, the defense system improves the network throughput and the service packet delivery rates, reduces the service packet drop rate, and increases the attacking packet drop rate.

7.2 Future Directions

The proposed prototype defense system protects single service from the saturated flooding attacks. The guard nodes can filter out the malicious traffic and clean up the region between the guard nodes and the service provider. If a network-wide protection or back tracing the DDoS attacks is needed, a more complicated, distributed and dynamic

model of defense system is the direction of the future research. One possible solution is to insert more defense nodes. These defense nodes can group on mesh overlay or form several defense lines. Several small defense systems can work together and share the protection workload. How multiple groups of defense nodes collaborate efficiently and safely will be a remarkable subject for the future work. Different network may require different architecture or mechanism. A large amount of testing and verifying is required before the best architecture is realized for a specific network.

7.2.1 Multi-tier Architecture

Proposed defense system is composed of two tiers, which are a service provider and a layer of guard nodes. A multi-tier architecture may provide a better protection against the penetration attacks, and provide a quicker response and backtracking [134, 171]. But multi-tier architecture is more complicated and introduces more overhead. How the layers are deployed and how they coordinate and adapt on the changes will be the future task in this research field. To maintain a multi-tier system in a wireless mobile environment leads to higher system overhead than it from wired networks [134, 171]. It is also important to find a mechanism with the best performance.

7.2.2 Mesh Overlay Architecture

A group of guard nodes of different services or a group of public security nodes in a network can collaborate and form a distributed overlay-based defense system to share the information and the security workload. This model makes it possible to push the defense and the back-trace towards the attackers. If there are more cooperated defense

nodes scattered in the network, the whole defense system may meet and filter the attack traffic earlier and protect larger network region. An efficient management mechanism is needed to balance the workload with the minimum system overhead. A powerful and efficient authorization system is necessary to eliminate security threats. An inter-group protocol is required for group handshaking and dynamic adaptation. Any complicated solution is not welcome by wireless networks, so a large amount of testing and verifying is needed to solve the dilemma of performance and complexity.

7.3 Summery

Overall, this research proposes a novel and practical defense system, which significantly mitigates DDoS flooding attacks on the service and the network. The proposed defense system does not modify the existing AODV routing protocol or require cooperation from any non-system node. So the proposed defense system is very easy to deploy. The experiment results demonstrate the defense system can mitigate DDoS attacks targeting a specific service. When the network is under a data flooding attack, the defense system improves the network throughput and the service packet delivery rates, reduces the service packet drop rate, and blocks all attacking packets.

On the other hand, multi-tier architecture and mesh overlay architecture are two future research directions. They provide better and wider protection, but a large amount of testing and verifying is required to find out the tradeoff point of best performance and least complexity.

REFERENCES*

1. Ricochet-Team, Internet Worms: Self-spreading Malicious Programs. 2003, White Paper: http://www.mcafee.com/us/local_content/white_papers/wp_ricochetbriefworms.pdf.
2. CERT, Computer Emergency Response Team. 1988-, CERT: http://www.cert.org/.
3. CERT, CERT/CC Statistics 1988-2005. 2005, CERT: http://www.cert.org/stats/cert_stats.html.
4. Tony Krone, Hacking Motives. 2005, Australian Institute of Criminology: http://www.aic.gov.au/publications/htcb/htcb006.pdf.
5. www.allaboutmarketresearch.com, Internet Growth and Stats, w.i.c. IDC, Editor. 2006: http://www.allaboutmarketresearch.com/internet.htm.
6. George H. Forman and John Zahorjan, The Challenges of Mobile Computing. IEEE Computer, 1994. 27(6): p. 38--47.
7. Samba Sesay, Zongkai Yang and Jianhua He, A Survey on Mobile Ad Hoc Wireless Network. Information Technology Journal, 2004. 3(2): p. 168--175.
8. Ian F. Akyildiz, Weilian Su, Yogesh Sankarasubramaniam and Erdal Cayirci, Wireless Sensor Networks: a Survey. Computer Networks, 2002. 38: p. 393--422.
9. David Karig and Ruby Lee, Remote Denial of Service Attacks and Countermeasures, T.R. CE-L2001-002, Editor. 2001, Princeton University Department of Electrical Engineering
10. Panix, Panix Under Attack. 1996, Panix: http://www.panix.com/press/synattack.html.
11. Jelena Mirkovic, Sven Dietrich, David Dittrich and Peter Reiher., Internet Denial of Service: Attack and Defense Mechanisms. 2005: Prentice Hall PTR 400.
12. Vern Paxson, An Analysis of Using Reflectors for Distributed Denial-of-Service Attacks. ACM Computer Communications Review (CCR), 2001. 31(3): p. 38--47.
13. Marc Zissman, Intrusion Detection Attacks Database. 1999: http://www.ll.mit.edu/IST/ideval/docs/1999/attackDB.html.
14. Seth Fogie and Cyrus Peikari, The Ingredients to ARP Poison 2002, Prentice Hall PTR: http://www.governmentsecurity.org/articles/TheIngredientstoARPPoison.php.
15. CERT, Teardrop and Land. 1997, CERT: http://www.cert.org/advisories/CA-1997-28.html.
16. CERT, Email Bombing and Spamming. 2002, CERT: http://www.cert.org/tech_tips/email_bombing_spamming.html.

* All web pages accessed on July 18th, 2006.

17. Jonathan Lemon. Resisting SYN Flood DoS Attacks with a SYN Cache. in BSDCon 2002 2002: USENIX Association. p. 89--97.
18. CERT, Denial-of-Service Attack Via Ping. 1996, CERT: http://www.cert.org/advisories/CA-1996-26.html.
19. CERT, Smurf Attack. 1998, CERT: http://www.cert.org/advisories/CA-1998-01.html.
20. Paul A. Watson. Slipping in the Window: TCP Reset Attacks. in CanSecWest 2004.
21. CERT, UDP Packet Storm. 1996, CERT: http://www.cert.org/advisories/CA-1996-01.html.
22. Kevin J. Houle and George M. Weaver, Trends in Denial of Service Attack Technology. 2001, CERT: http://www.cert.org/archive/pdf/DoS_trends.pdf.
23. Stephen Specht and Ruby Lee. Distributed Denial of Service: Taxonomies of Attacks, Tools, and Countermeasures. in the 17th Int'l Conf. Parallel and Distributed Computing Systems. 2004. p. 536--543.
24. CERT, Distributed Denial of Service Tools. 1999, CERT: http://www.cert.org/incident_notes/IN-99-07.html.
25. CERT, "mstream" Distributed Denial of Service Tool. 2000, CERT: http://www.cert.org/incident_notes/IN-2000-05.html.
26. David Dittrich, Analysis of Stacheldraht. 1999: http://staff.washington.edu/dittrich/misc/stacheldraht.analysis.
27. "The Honeynet Project & Research Alliance", Know your Enemy: Tracking Botnets. 2005. http://www.honeynet.org/papers/bots/.
28. Amrit Williams and Jay Heiser, Protect Your PCs and Servers From the Botnet Threat. 2004, Gartner, Inc.
29. CERT, Similar Attacks Using Various RPC Services. 1999, CERT: http://www.cert.org/incident_notes/IN-99-04.html.
30. Sophos Plc, W32/Agobot-LI. http://www.sophos.com/virusinfo/analyses/w32agobotli.html, 2004.
31. Allen Householder, Managing the Threat of Denial-of-Service Attacks. 2001, CERT: http://www.cert.org/archive/pdf/Managing_DoS.pdf.
32. Jelena Mirkovic and Peter Reiher, A Taxonomy of DDoS Attack and DDoS Defense Mechanisms. ACM SIGCOMM Computer Communication Review 2004. 34(2): p. 39--53.
33. Ahsan Habib, Mohamed Hefeeda and Bharat Bhargava. Detecting Service Violations and DoS Attacks. in The 10th Annual Network and Distributed System Security Symposium 2003. San Diego, California. p. 177--189.
34. Mun Choon Chan, Yow-Jian Lin and Xin Wang. A Scalable Monitoring Approach for Service Level Agreements Validation. in International Conference on Network Protocols (ICNP). 2000: IEEE Computer Society. p. 37--48.
35. R. Caceres, N.G. Duffield, J. Horowitz and D. Towsley, Multicast-Based Inference of Network-Internal Loss Characteristics IEEE Transactions on Information Theory, 1999. 45(7): p. 2462--2480.
36. N.G. Duffield, F. Lo Presti, V. Paxson and D. Towsley. Inferring Link Loss Using Striped Unicast Probes. in IEEE INFOCOM 2001. 2001. Anchorage, Alaska. p. 915--923.

37. Ahsan Habib, Maleq Khan and Bharat Bhargava, Edge-to-Edge Measurement-based Distributed Network Monitoring. Computer Networks: The International Journal of Computer and Telecommunications Networking 2004. 44(2): p. 211--233.
38. Steven Bellovin, Marcus Leech and Tom Taylor, ICMP Traceback Messages. 2000, IETF: http://www3.ietf.org/proceedings/01aug/I-D/draft-ietf-itrace-00.txt.
39. Alex C. Snoeren, Craig Partridge, Luis A. Sanchez, Christine E. Jones, Fabrice Tchakountio, Stephen T. Kent and W. Timothy Strayer. Hash-Based IP Traceback. in 2001 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications. 2001. San Diego, CA: ACM. p. 3--14.
40. Hal Burch and Bill Cheswick. Tracing Anonymous Packets to Their Approximate Source. in The 14th USENIX Conference on System Administration. 2000. New Orleans, Louisiana USENIX Association. p. 319--328.
41. Kihong Park and Heejo Lee. On the Effectiveness of Probabilistic Packet Marking for IP Traceback Under Denial of Service Attack. in IEEE INFOCOM '01. 2001. Anchorage, Alaska: IEEE. p. 338--347.
42. Michael R. Lyu and Lorrien K. Y. Lau. Firewall Security: Policies, Testing and Performance Evaluation. in 24th International Computer Software and Applications Conference (COMPSAC 2000). 2000. Taipei, Taiwan: IEEE Computer Society. p. 116--121.
43. Paul Ferguson and Daniel Senie, Network Ingress Filtering: Defeating Denial of Service Attacks Which Employ IP Source Address Spoofing. 2000, IETF: http://www.rfc-archive.org/getrfc.php?rfc=2827.
44. Sans, Egress Filtering v 0.2. 2000, SANS: http://www.sans.org/y2k/egress.htm.
45. Kihong Park and Heejo Lee. On the Effectiveness of Route-Based Packet Filtering for Distributed DoS Attack Prevention in Power-Law Internets. in the 2001 conference on Applications, technologies, architectures, and protocols for computer communications. 2001. San Diego, California: ACM Press p. 15--26
46. NIST, Mobile Ad Hoc Networks (MANETs). 2001, NIST: http://w3.antd.nist.gov/wahn_mahn.shtml.
47. Intel, Understanding Wi-Fi and WiMAX as Metro-Access Solutions. 2004, INTEL: http://www.intel.com/business/bss/industry/government/wimaxandmeshwhitepaper.pdf.
48. Qianhui Liang, Lakshmi N. Chakarapani, Stanley Y.W. Su, Raman N. Chikkamagalur and Herman Lam, A Semi-automatic Approach to Composite Web Services Discovery, Description and Invocation. International Journal of Web Services Research, 2004. 1(4): p. 64--89.
49. Françoise Sailhan and Valérie Issarny. Scalable Service Discovery for MANET. in the 3rd IEEE International Conference on Pervasive Computing and Communications 2005.
50. Stephen S. Yau, Yu Wang and Dazhi Huang. Middleware Support for Embedded Software with Multiple QoS Properties for Ubiquitous Computing Environments. in the 8th IEEE International Workshop on Object-oriented Real-time Dependable Systems 2003. Guadalajara, Mexico. p. 250--256.

51. Dijiang Huang, Amit Sinha and Deep Medhi. A Double Authentication Scheme To Detect Impersonation Attack In Link State Routing Protocols. in IEEE International Conference on Communications (ICC). 2003. Anchorage, Alaska. p. 1723--1727.
52. Tom Karygiannis and Les Owens, Wireless Network Security. 2003, NIST: http://csrc.nist.gov/publications/nistpubs/800-48/NIST_SP_800-48.pdf.
53. Lidong Zhou and Zygmunt J. Haas, Securing Ad Hoc Networks. IEEE Network Magazine, 1999. 13(6): p. 24--30.
54. Gretchen H. Lynn, ROMR: Robust Multicast Routing in Mobile Ad-Hoc Networks, in School of Arts and Sciences. 2003, University of Pittsburgh. p. 133.
55. Shree Murthy and J. J. Garcia-Luna Aceves. A Routing Protocol for Packet Radio Networks. in Mobile Computing and Networking. 1995. Berkeley, CA: ACM. p. 86--95.
56. Leonard Kleinrock and Kent Stevens, Fisheye: A Lenslike Computer Display Transformation. 1971, UCLA Technical report.
57. Charles E. Perkins and Pravin Bhagwat. Highly Dynamic Destination-Sequenced Distance Vector Routing (DSDV) for Mobile Computers. in '94 ACM Conference on Communications Architectures, Protocols and Applications. 1994. London, U.K. p. 234--244.
58. Thomas Clausen, Philippe Jacquet, Anis Laouiti, Pascale Minet, Paul Muhlethaler, Amir Qayyum, Laurent Viennot and INRIA Rocquencourt, Optimized Link State Routing Protocol (OLSR). 2001, IETF Internet draft.
59. Ching-Chuan Chiang, Hsiao-Kuang Wu, Winston Liu and Mario Gerla. Routing in Clustered Multihop, Mobile Wireless Networks with Fading Channel. in IEEE Singapore International Conference on Networks. 1997. Singapore: IEEE. p. 197--211.
60. David B. Johnson and David A. Maltz, Dynamic Source Routing in Ad Hoc Wireless Networks, in Mobile Computing, I.a. Korth, Editor. 1996, Kluwer Academic Publishers. p. 153--181.
61. Charles Perkins, Elizabeth Royer, Samir Das, Ad Hoc On Demand Distance Vector (AODV) Routing. IETF Internet Draft, 2003. http://www3.ietf.org/proceedings/03mar/I-D/draft-ietf-manet-aodv-13.txt.
62. Ionut D. Aron and Sandeep K. S. Gupta, On the Scalability of On-demand Routing Protocols for Mobile Ad Hoc Networks: an Analytical Study. Journal of Interconnection Networks, 2001. 2(1): p. 5--29.
63. Giovanni Vigna, Sumit Gwalani, Kavitha Srinivasan, Elizabeth M. Belding-Royer and Richard A. Kemmerer. An Intrusion Detection Tool for AODV-based Ad hoc Wireless Networks. in 20th Annual Computer Security Applications Conference (ACSAC'04). 2004: IEEE Computer Society p. 16--27.
64. Josh Broch, David A. Maltz, David B. Johnson, Yih-Chun Hu and Jorjeta Jetcheva. A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols. in 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking 1998. Dallas, Texas, United States ACM Press p. 85--97.
65. Elizabeth M. Royer and Charles E. Perkins. Multicast Operation of the Ad-hoc On-Demand Distance Vector Routing Protocol. in ACM/IEEE Intl. Conference

on Mobile Computing and Networking (MOBICOM), . 1999. Seattle, WA. p. 207--218.
66. Jin-Man Kim and Jong-Wook Jang. Performance Evaluation of AODV-based Power-Aware Routing Protocol in Mobile Ad Hoc Networks. in Networks and Communication Systems. 2005. Krabi, Thailand. p. 464--067.
67. Shinji Motegi and Hiroki Horiuchi, Proposal on AODV-based Multipath Routing Protocol for Mobile Ad Hoc Networks. 2004, KDDI Laboratories, Inc.: http://www.unl.im.dendai.ac.jp/INSS2004/INSS2004_papers/PosterPresentations/P3.pdf.
68. Shivanajay Marwaha, Chen Khong Tham and Dipti Srinivasan. Mobile Agents Based Routing Protocol for Mobile Ad Hoc Networks. in IEEE Globecom. 2002: IEEE Computer Society Press. p. 17--21.
69. Vincent D. Park and M. Scott Corson, Temporally-ordered Routing Algorithm (TORA) Version 1: Functional Specification. 2001: IETF Internet draft.
70. Eli M. Gafni and Dimitri P. Bertsekas. Distributed Algorithms for Generating Loop-free Routes in Networks With Frequently Changing Topology. in IEEE Transactions on Communications. 1981. p. 11--18.
71. Spilios Giannoulis, Christos Antonopoulos, Evangelos Topalis and Stavros Koubias, ZRP Versus DSR and TORA: A Comprehensive Survey on ZRP Performance. 2003: ETFA 2005.
72. Raghupathy Sivakumar, Prasun Sinha and Vaduvur Bharghavan. CEDAR: A Core-Extraction Distributed Ad hoc Routing Algorithm. in IEEE INFOCOMM '99. 1999. New York, NY: IEEE.
73. Rohit Dube, Cynthia D. Rais, Kuang-Yeh Wang and Satish K. Tripathi, Signal Stability-Based Adaptive Routing (SSA) for Ad-Hoc Mobile Networks. IEEE Personal Communication, 1996. 4(1): p. 36--45.
74. Zygmunt J. Haas, Marc R. Pearlm and prince Samar, The Zone Routing Protocol (ZRP) for Ad Hoc Networks. 1997, IETF Internet draft.
75. Venugopalan Ramasubramanian, Zygmunt J. Haas and Emin G¨un Sirer. SHARP: A Hybrid Adaptive Routing Protocol for Mobile Ad Hoc Networks. in the 4th ACM International Symposium on Mobile Ad hoc Networking & Computing 2003. Annapolis, Maryland: ACM Press. p. 303--314.
76. Silvia Giordano and Ivan Stojmenovic, Position Based Routing Algorithms For Ad Hoc Networks: A Taxonomy. 2001.
77. Young-Bae Ko and Nitin H. Vaidya. Location-aided Routing (LAR) in Mobile Ad Hoc Networks. in the 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking 1998. Dallas, Texas, United States ACM Press p. 66--75.
78. Brad Karp and H. T. Kung, GPSR: Greedy Perimeter Stateless Routing for Wireless Networks, in Mobile Computing and Networking. 2000. p. 243--254.
79. Shigang Chen and Randy Chow. A New Perspective in Defending Against DDoS in 10th IEEE International Workshop on Future Trends of Distributed Computing Systems (FTDCS 2004). 2004. Suzhou, China: IEEE Computer Society. p. 186--190.

80. Guangyu Pei, Mario Gerla and Xiaoyan Hong. LANMAR: Landmark Routing for Large Scale Wireless Ad Hoc Networks with Group Mobility. in IEEE/ACM MobiHOC 2000. 2000. Boston, MA. p. 11--18.
81. Matthias Transier, Holger Fübler, Jörg Widmer, Martin Mauve and Wolfgang Effelsberg, A Hierarchical Approach to Position-Based Multicast for Mobile Ad-hoc Networks. 2004, Department of Computer Science, University of Mannheim: http://bibserv7.bib.uni-mannheim.de/madoc/volltexte/2004/725/pdf/Transier2004a.pdf.
82. Anthony D. Wood and John A. Stankovic, Denial of Service in Sensor Networks. Computer 2002. 35(10): p. 54--62.
83. Randall K. Nichols and Panos C. Lekkas, Wireless Security: Models, Threats, and Solutions 1ed. 2002: McGraw-Hill Professional. p. 657.
84. Julie Schuller, Understanding Wireless LAN Technology and Its Security Risks. 2003, GIAC: http://www.giac.org/practical/GSEC/Julie_Schuller_GSEC.pdf.
85. Wenyuan Xu, Wade Trappe, Yanyong Zhang and Timothy Wood. The Feasibility of Launching and Detecting Jamming Attacks in Wireless Networks. in 6th ACM International Symposium on Mobile Ad Hoc Networking and Computing 2005. Urbana-Champaign, IL, USA: ACM Press p. 46--57.
86. Jeremy Blum and Azim Eskandarian, The Threat of Intelligent Collisions. IT Professional, 2004. 6(1): p. 24--29.
87. Adam Stubblefield, John Ioannidis and Aviel D. Rubin, Using the Fluhrer, Mantin, and Shamir Attack to Break WEP. 2002, Internet Society (ISOC) http://www.isoc.org/isoc/conferences/ndss/02/proceedings/papers/stubbl.pdf.
88. Mohammad Al-Shurman, Seong-Moo Yoo and Seungjin Park. Black Hole Attack in Mobile Ad Hoc Networks. in the 42nd Annual Southeast Regional Conference. 2004. Huntsville, Alabama, USA: ACM. p. 96--97.
89. Jonathan Katz, Efficient Cryptographic Protocols Preventing "Man-in-the-Middle" Attacks. 2002, PhD Dissertation, Columbia University.
90. Sanjay Ramaswamy, Huirong Fu, Manohar Sreekantaradhya, John Dixon and Kendall Nygard. Prevention of Cooperative Black Hole Attack in Wireless Ad Hoc Networks. in International Conference on Wireless Networks 2003. 2003. p. 570--575.
91. Abbie Barbir, Sandra Murphy and Yibin Yang, Generic Threats to Routing Protocols 2004, IETF Internet draft. http://www.ietf.org/internet-drafts/draft-ietf-rpsec-routing-threats-07.txt.
92. Ping Yi, Zhoulin Dai, Yiping Zhong and Shiyong Zhang. Resisting Flooding Attacks in Ad Hoc Networks in International Conference on Information Technology: Coding and Computing (ITCC'05). 2005. Las Vegas, Nevada, USA: IEEE Computer Society. p. 657--662.
93. Mike Just, Evangelos Kranakis and Tao Wan. Resisting Malicious Packet Dropping in Wireless Ad Hoc Networks. in ADHOCNOW'03. 2003. Montreal, Canada.
94. Yih-Chun Hu, Adrian Perrig and David B. Johnson. Rushing Attacks and Defense in Wireless Ad Hoc Network Routing Protocols. in 2nd ACM Wireless Security (WiSe'03). 2003. p. 30--40.

95. Marco Conti, Enrico Gregori and Gaia Maselli. Towards Reliable Forwarding for Ad Hoc Networks. in Personal Wireless Communications, IFIP-TC6 8th International Conference, PWC 2003. Venice, Italy: Springer. p. 790--804.
96. Refik Molva and Pietro Michiardi. Security in Ad Hoc Networks. in Personal Wireless Communications, IFIP-TC6 8th International Conference. 2002. Venice, Italy: Springer. p. 756--775.
97. Ion Stoica, Daniel Adkins, Shelley Zhuang, Scott Shenker and Sonesh Surana. Internet Indirection Infrastructure. in ACM SIGCOMM Conference 2002. Pittsburgh, PA, USA. p. 73--88.
98. Loukas Lazos, Radha Poovendran, C. Meadows, P. Syverson and L.W. Chang. Preventing Wormhole Attacks on Wireless Ad Hoc Networks: A Graph Theoretic Approach. in IEEE Wireless Communications and Networking Conference. 2005. p. 1193--1199.
99. T. Charles Clancy, Security Review of the Light-Weight Access Point Protocol. 2005, IETF CAPWAP Working Group.
100. Lee Lawson, Session Hijacking Packet Analysis. 2005, SecurityDocs.com Report.
101. ISO/IEC JTC, ISO/IEC JTC 1/SC 25 INTERCONNECTION OF INFORMATION TECHNOLOGY EQUIPMENT Secretariat: Germany (DIN). 1996, ISO/IEC: http://www.t10.org/ftp/t10/document.96/96-210r0.pdf.
102. Jolyon Clulow, The Design and Analysis of Cryptographic Application Programming Interfaces for Security Devices. 2003, University of Natal. p. 133.
103. Kenneth Graf, Addressing Challenges in Application Security. 2005, A Watchfire White Paper. http://www.watchfire.com.
104. Black Box Corp., Network Security, A White Paper. 2003. http:// www.blackbox.com/Tech_Support/White-Papers/Network-Security2.pdf.
105. Jeffrey W. Humphries and Martin C. Carlisle, Introduction to Cryptography. ACM Journal of Educational Resources in Computing (JERIC) 2002. ISSN:1531-4278. 2(3): p. 2.
106. SungJun Min, A Study on the Security of NTRUSign Digital Signature Scheme. 2004, Master Thesis in Information and Communications University, Korea.
107. Xiaoyun Wang, Dengguo Feng, Xuejia Lai and Hongbo Yu, Collisions for Hash Functions MD4, MD5, HAVAL-128 and RIPEMD. 2004, Cryptology ePrint Archive.
108. Charlie Kaufman, Radia Perlman and Mike Speciner, Network Security Private Communication in a Public World. 2002: Prentice Hall PTR. p. 752.
109. Sean Convery, Darrin Miller and Sri Sundaralingam, Cisco SAFE: Wireless LAN Security in Depth 2003, CISCO Whitepaper.
110. Barbara Guttman, Computer Security Considerations in Federal Procurements: A Guide for Procurement Initiators, Contracting Officers and Computer Security Officials. 1992, NIST Special Publication 800-4.
111. Harold W. Fletcher, Kevin Richardson, Martin C. Carlisle and J.A. Hamilton Jr. Simulation Experimentation with Secure Overlay Services. in Summer Computer Simulation Conference. 2005. Philadelphia, Pa.
112. Karthik Lakshminarayanan, Daniel Adkins, Adrian Perrig and Ion Stoica. Taming IP Packet Flooding Attacks. in 2nd ACM Workshop on Hot Topics in Networks. 2003. Cambridge, MA: ACM Press p. 45--50.

113. Michael Collins and Michael K. Reiter. An Empirical Analysis of Target-Resident DoS Filters in 2004 IEEE Symposium on Security and Privacy. 2004. Oakland, California, USA: IEEE Computer Society. p. 103--114.
114. Jelena Mirkovic, Gregory Prier and Peter Reiher. Attacking DDoS at the Source. in the 10th IEEE International Conference on Network Protocols. 2002: IEEE Computer Society. p. 312--321.
115. Lidong Zhou, Fred B. Schneider and Robbert van Renesse, COCA: A Secure Distributed On-line Certification Authority. ACM Transactions on Computer Systems, 2002. 20(4): p. 329--368.
116. B. Bencsáth and I. Vajda. Protection Against DDoS Attacks Based On Traffic Level Measurements. in International Symposium on Collaborative Technologies and Systems. 2004. San Diego, CA.
117. Jussipekka Leiwo, Tuomas Aura and Pekka Nikander. Towards Network Denial Of Service Resistant Protocols. in 15th International Information Security Conference 2000. Beijing, China. p. 301--310.
118. Stephen S. Yau, Yu Wang and Fariaz Karim. Development of Situation-Aware Application Software for Ubiquitous Computing Environment. in 26th International Computer Software and Applications Conference on Prolonging Software Life: Development and Redevelopment 2002: IEEE Computer Society. p. 233--238.
119. Jun Li, Minho Sung, Jun (Jim) Xu and Li (Erran) Li. Large-Scale IP Traceback in High-Speed Internet: Practical Techniques and Theoretical Foundation. in IEEE Symposium on Security and Privacy 2004. Oakland, California, USA. 115.
120. Basheer Duwairi, Anirban Chakrabarti and Govindarasu Manimaran. An Efficient Probabilistic Packet Marking Scheme for IP Traceback. in 3rd IFIP-TC6 Networking. 2004. Athens, Greece. p. 1263--1269.
121. John Ioannidis and Steven M. Bellovin. Implementing Pushback: Router-Based Defense Against DDoS Attacks. in Network and Distributed System Security Symposium. 2002. San Diego, CA. p. 79--86.
122. Stefan Savage, David Wetherall, Anna Karlin and Tom Anderson. Practical Network Support for IP Traceback. in the 2000 ACM SIGCOMM Conference. 2000. Stockholm, Sweden. p. 295--306.
123. Drew Dean, Matt Franklin and Adam Stubblefield, An Algebraic Approach to IP Traceback. Information and System Security, 2002. 5(2): p. 119--137.
124. Dan Schnackenberg, Kelly Djahandari and Dan Sterne. Infrastructure for Intrusion Detection and Response. in DARPA Information Survivability Conference and Exposition. 2000. p. 1003--1011.
125. Ricardo Puttini, Jean-Marc Percher, L.Mé and Rafael de Sousa. A Fully Distributed IDS for MANET. in the 9th IEEE Symposium on Computers and Communications 2004. New Jersey, USA: IEEE. p. 331--338.
126. Patrick Albers, Olivier Camp, Jean-Marc Percher, Bernard Jouga, Ludovic M´e and Ricardo Puttini. Security in Ad Hoc Networks: a General Intrusion Detection Architecture Enhancing Trust Based Approaches. in 1st International Workshop on Wireless Information Systems. 2002.
127. Abraham Yaar, Adrian Perrig and Dawn Song. SIFF: A Stateless Internet Flow Filter to Mitigate DDoS Flooding Attacks. in the IEEE Security and Privacy

Symposium. 2004. Philadelphia, Pennsylvania, USA ACM Press New York, NY, USA p. 241--252.
128. Tom Anderson, Timothy Roscoe and David Wetherall, Preventing Internet Denial-of-Service with Capabilities. ACM SIGCOMM Computer Communication Review, 2004. 34(1): p. 39--44.
129. Tuomas Aura, Pekka Nikander and Jussipekka Leiwo, DOS-Resistant Authentication with Client Puzzles. Lecture Notes in Computer Science, 2001. 2133: p. 170--178.
130. Brent Waters, Ari Juels, J. Alex Halderman and Edward W. Felten. New Client Puzzle Outsourcing Techniques for DoS Resistance. in 11th ACM conference on Computer and communications security. 2004. Washington DC, USA: ACM Press, p. 246--256.
131. Dan J. Bernstein, SYN cookies. 1996: http://cr.yp.to/syncookies.html.
132. Angelos D. Keromytis, Vishal Misra and Daniel Rubenstein. Using Overlays to Improve Network Security. in SPIE ITCom Conference on Scalability and Traffic Control in IP Networks II. 2002.
133. Robert Stone. Centertrack: An ip Overlay Network for Tracking DoS Floods. in 9th USENIX Security Symposium. 2000.
134. Angelos D. Keromytis, Vishal Misra and Dan Rubenstein. SOS: Secure Overlay Services. in ACM SIGCOMM'02. 2002. Pittsburgh, PA.
135. David G. Andersen. Mayday: Distributed Filtering for Internet Services. in 4th USENIX Symposium on Internet Technologies and Systems (USITS). 2003. Seattle, Washington.
136. Harold W. Fletcher, Kevin Richardson, Martin C. Carlisle and J.A. Hamilton Jr. Evaluating Secure Overlay Services Through OPNET Simulation. in SCS Spring Simulation Multiconference. 2005. San Diego, CA.
137. Jiejun Kong and Xiaoyan Hong. ANODR: ANonymous On Demand Routing with Untraceable Routes for Mobile Ad-hoc Networks. in the 4th ACM International Symposium on Mobile Ad Hoc Networking & Computing 2003. Annapolis, Maryland, USA ACM Press. p. 291--302.
138. Alf Zugenmaier and Adolf Hohl. Anonymity for Users of Ubiquitous Computing. in 2nd Workshop on Security in Ubiquitous Computing 2003. UBICOMP 2003, Seattle, Washington, USA. p. 06.
139. Michael Atighetchi, Partha Pal, Franklin Webber and Christopher Jones. Adaptive Use of Network-Centric Mechanisms in Cyber-Defense. in Sixth IEEE International Symposium on Object-Oriented Real-Time Distributed Computing. 2003: IEEE Computer Society. p. 183.
140. Liang Wei, Yu Haibin, Che Chang and Bai Jieyin. Research on Simulation of Wireless Sensor Network in 16th IFAC World Congress. Prague.
141. OPNET: http://www.opnet.com/.
142. Virtual InterNetwork Testbed (VINT Project), Ns-2. 1995, nsnam: http://nsnam.isi.edu/nsnam/index.php/Main_Page.
143. Stuart Kurkowski, Tracy Camp and Michael Colagrosso, MANET Simulation Studies: The Current State and New Simulation Tools. 2005, The Colorado School of Mines, Golden, Colorado.

144. James D. Box, Adam Hathcock, Alan Hunt, Maj. J.L. Humphries and J.A. Hamilton Jr. Simulation Strategic Firewall Placement. in SCS Spring Simulation Multiconference. 2005. San Diego, CA.
145. Gilberto Flores Lucio, Marcos Paredes-Farrera, Emmanuel Jammeh, Martin Fleury and Martin J. Reed, OPNET Modeler and Ns-2: Comparing the Accuracy of Network Simulators for Packet-Level Analysis using a Network Testbed. WSEAS Transactions on Computers, 2003. 2(3): p. 700--707.
146. D. Anick, D. Mitra and M.M. Sondhi, Stochastic Theory of A Data-handling System with Multiple Sources. The Bell System Technical Journal, 1982. 61(8): p. 1871--1894.
147. Tak Kin Yung, Jay Martin, Mineo Takai and Rajive Bagrodia. Integration of Fluid-based Analytical Model with Packet-Level Simulation for Analysis of Computer Networks. in SPIE. 2001. p. 130--143.
148. Anlu Yan and Wei-Bo Gong. Time-driven Fluid Simulation for High-speed Networks. in IEEE Transactions on Information Theory. 1999. p. 1588--1599.
149. Benyuan Liu, Yang Guo, Jim Kurose, Don Towsley and Weibo Gong. Fluid Simulation of Large Scale Networks: Issues and Tradeoffs. in International Conference on Parallel and Distributed Processing Techniques and Applications. 1999. Las Vegas, NV. p. 2136--2142.
150. Lee Breslau, Deborah, Estrin, Kevin Fall, Sally Floyd, John, Heidemann, Ahmed, Helmy, Polly Huang, Steven, McCanne, Kannan, Varadhan, Ya Xu and Haobo Yu, Advances in Network Simulation. Computer, 2000. 33(5): p. 59--67.
151. Björn Wiberg, Porting AODV-UU Implementation to ns-2 and Enabling Trace-based Simulation, in Information Technology Department of Computer Systems. 2002, Uppsala University: Uppsala. p. 104.
152. Oskar Wibling, Ad Hoc Routing Protocol Validation, in Department of Information Technology. 2005, Uppsala University: Uppsala. p. 102.
153. Ian D. Chakeres and Elizabeth M. Belding-Royer. Transparent Influence of Path Selection in Heterogeneous Ad hoc Networks. in 15th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC). 2004. Barcelona, Spain. p. 885--889.
154. Madanlal Musuvathi, David Park, Andy Chou, Dawson R. Engler and David L. Dill. CMC: A Pragmatic Approach to Model Checking Real Code. in 5th USENIX Symposium on Operating Systems Design and Implementation (OSDI). 2002. Boston, MA. p. 75--88.
155. Charles Perkins. Ad Hoc On Demand Distance Vector (AODV) Routing. in 2nd IEEE Workshop on Mobile Computing Systems and Applications. 1999. New Orleans, LA,: IEEE. p. 90--100.
156. CERT, Denial of Service Attacks. 2001, CERT: http://www.cert.org/tech_tips/denial_of_service.html.
157. Lidong Zhou, Survivability: Beyond Fault Tolerance and Cryptography. 2004, Microsoft Research report.
158. Fred B. Schneider and Lidong Zhou, Distributed Trust: Supporting Fault-tolerance and Attack-tolerance. 2004, Cornell University Technical Report 2004.1924.

159. Sherif M. Khattab, Chatree Sangpachatanarukx, Rami Melhem, Daniel Moss and Taieb Znati, Proactive Server Roaming for Mitigating Denial-of-Service Attacks. Journal of Systems and Software 2004. 73(1): p. 15--29.
160. J. Jones, Distributed Denial of Service Attacks: Defenses, A Special Publication. 2000, Global Integrity Technical Report.
161. Aleksandar Kuzmanovic, Dan Dumitriu, Ed Knightly, Ion Stoica and Willy Zwaenepoel. Denial-of-Service Resilience in Peer-to-Peer File Sharing Systems. in ACM SIGMETRICS 2005. 2004. Banff, Alberta, Canada p. 38--49.
162. Cheng Jin, Haining Wang and Kang G. Shin. Hop-Count Filtering: An Effective Defense Against Spoofed DDoS Traffic. in 10th ACM International Conference on Computer and Communications Security (CCS). 2003. New York: ACM Press. p. 30--41.
163. Angelos Stavrou, Debra L. Cook, William G. Morein, Angelos D. Keromytis, Vishal Misra and Dan Rubenstein, WebSOS: An Overlay-based System for Protecting Web Servers from Denial of Service Attacks. Computer Networks: The International Journal of Computer and Telecommunications Networking, 2005. 48(5): p. 781--807.
164. Victor S Miller. Use of Elliptic Curves in Cryptography. in Advances in Cryptology---CRYPTO 85 1986 Santa Barbara, California, United States: Springer-Verlag New York, Inc. . p. 417--426.
165. Neal Koblitz. Elliptic Curve Cryptosystems. in Mathematics of Computation. 1987. p. 203--209.
166. Kristin Lauter, The Elliptic Curve Digital Signature Algorithm (ECDSA), in IEEE Wireless Communications. 1999. p. 62--67.
167. Samir R. Das, Charles E. Perkins and Elizabeth M. Royer, Performance Comparison of Two On-demand Routing Protocols for Ad Hoc Networks. IEEE Personal Communications, 2001. 8(1): p. 171--178.
168. Sung-Ju Lee, Elizabeth M. Belding-Royer and Charles E. Perkins, Ad Hoc On-demand Distance-vector Routing Scalability. ACM SIGMOBILE Mobile Computing and Communications Review, 2002. 6(3): p. 94--95.
169. Dhaval Gada, Rajat Gogri, Punit Rathod, Zalak Dedhia, Nirali Mody, Sugata Sanyal and Ajith Abraham, A Distributed Security Scheme for Ad Hoc Networks. ACM Crossroads, Special Issue on Computer Security, 2004. 11.
170. Charles E. Perkins, Elizabeth M. Belding-Royer and Samir Das, RFC3561: Ad hoc On-Demand Distance Vector (AODV) Routing 2003, Network Working Group RFC.
171. Jun Wang and Andrew. A. Chien, Using Overlay Networks to Resist Denial-of-Service Attacks. 2003, University of California, San Diego: Technical report.
172. D. Patrick Allen, Nagui Rouphail, Joseph E. Hummer and Joseph S. Milazzo II, Operational Analysis of Uninterrupted Bicycle Facilities, in Transportation Research Record 1636. 1998: Washington, D.C. p. 29--36.

APPENDIX A

There is no AODV protocol specification modified, no functionality of regular packet sending/receiving of regular node changed, nor functionality of routing table management changed. Guard node functionality is modularized and encapsulated.

A.1 AODV-UU Agent class Modifications and Extension in ns/aodv-uu.h:

```
//XYU: For debugging message: Define the displaying node
type.
enum {
  COMMON,
  CLIENT,
  SERVER,
  GUARD,
  ATTACKER
};
// AODV-UU node agent definition
class AODVUU : public Agent
{//...
  // Defense system node extension.
  // node type.
```

```
  int               sim_node_type;
  // Shield System Filter.
  int               ShieldSys_flag;
  int               ShieldSys_providerID;   // Actual
server ID.
  int               ShieldSys_svrID;        // Pseudo
Server (network service) ID.
  int               ShieldSys_Cli_1;        // Simulation
client ID 1.
  int               ShieldSys_Cli_2;        // Simulation
client ID 2.
  int               ShieldSys_gd_1;         // guard node.
  int               ShieldSys_gd_2;         // guard node.
  int               ShieldSys_gd_3;         // guard node.
  int               ShieldSys_gd_4;         // guard node.
  int               ShieldSys_guard_IP;     // guard node
current IP address.
  // Legitimate client management.
  void              ShieldSys_push(struct in_addr
&cli_addr);  // Add a legitimate client.
  bool              ShieldSys_isLegal(struct in_addr
&cli_addr);// legitimacy checking.
  bool              ShieldSys_Filter(Packet *p);
// Guard filtering.
```

```
  void              ShieldSys_setAlarm();
// Network under Attack.
  void              ShieldSys_cleanAlarm();
// Attack is over.
  void              ShieldSys_forward(Packet *p);
// Guard tunneling.
  Packet           *Duplicate(Packet *p);
// Utility function.
  bool              ShieldSys_alarm_;
  static struct in_addr
ShieldSys_prior_list[PRIOR_LIST_SIZE];  // legitimate list.
  int               ShieldSys_index_;
};
```

A.2 Code Modifications in aodv_rreq.c:

```
//XYU: For debugging message: get current time.
//     The difference to the start of the system.
float getcurrTime(int startflag)
{
  static struct timeval start;
  struct timeval now;
  u_int32_t diff;
  if(startflag) {
```

```
    gettimeofday(&start, NULL);
  }
  gettimeofday(&now, NULL);
  diff = timeval_diff(&now, &start);
  return diff;
}

void NS_CLASS rreq_process()
{
  //...
  /* XYU: Guard node filter RREQ packets towards Svr
here.*/
  if(sim_node_type == GUARD && rreq_dest.s_addr ==
ShieldSys_svrID) {
    //XYU: forced the RREP to announce the path to the
service (PseudoSvrID).
    if (rreq_dest_seqno != 0) {
    // Modify the sequence number of the RREQ packet.
      if ((int32_t) this_host.seqno < (int32_t)
rreq_dest_seqno)
        this_host.seqno = rreq_dest_seqno;
      else if (this_host.seqno == rreq_dest_seqno)
        seqno_incr(this_host.seqno);
    }
```

```
    // Guard node returns RREP.
    // Reply with the newest Guard IP. Refer section
3.4.3.2 in the dissertation.
    rrep = rrep_create(0, 0, 0, rreq_dest,  // point to
PseudoSvrID.
                                this_host.seqno,
                                rev_rt->dest_addr,
                                MY_ROUTE_TIMEOUT);
    rrep_send(rrep, rev_rt, NULL, RREP_SIZE);
    return;
  }
//...
}
```

A.3 Code Modifications in aodv_socket.c :

```
void NS_CLASS aodv_socket_send()
{//...
    // Hack the RREQ_RATELIMIT, if it is an attacker, RREQ
counter will not increase.
    if(!rreqAttack_flag)
      num_rreq++;
}
```

A.4 Code Modifications and extension in ns/packet_input.cc:

```
void NS_CLASS processPacket()
{
  // ...
  // If it is A DATA packet, guard node will tunnel it
towards Svr
  // Defense system guard node filter non-legitimate
traffic under attack.
  if(sim_node_type == GUARD) {
    if(ShieldSys_Filter(p)) { // Return 0 to drop the
packet.
      return;
    }
  // Other illegitimate or attacking packets are dropped
here.
  }
}
```

A.5 New file of ns/aodv-shield.cc:

```
void NS_CLASS processPacket()
#include "../common/encap.h"
#include "aodv-uu.h"
static char recv_buf[RECV_BUF_SIZE];
```

```
// If the network is under attack, the guard nodes need
change the ip address periodically, refer section 3.4.3.2
in the dissertation.
void AODVUU::ShieldSys_IPHopping()
{
  if(ShieldSys_alarm_) {
    ShieldSys_guard_IP = ShieldSys_next_guardIP();
  }
}

// Lookup first.
// Loop inserting the legitimate client address.
void AODVUU::ShieldSys_push(struct in_addr &cli_addr)
{
}

// Legitimacy Checking.
// If the client is legitimate, return true;
// Otherwise, return false.
bool AODVUU::ShieldSys_isLegal(struct in_addr &cli_addr)
{
}
```

```
// Packet filter on the guard nodes.
// Return: 0 –- Regular network traffic. Let it go through
the routing.
//         1 –- A service related packet. Process it.
bool AODVUU::ShieldSys_Filter(Packet *p)
{
  struct hdr_ip *ih = HDR_IP(p);
  struct in_addr daddr;
  struct in_addr saddr;
  daddr.s_addr = ih->daddr();
  saddr.s_addr = ih->saddr();
  if(daddr.s_addr != ShieldSys_svrID) { // Not a packet to
the service provider.
    return false;
  }
  // This is a service client.
  // If the network is under attack, drop all illegitmate
or attack packets.
  // Otherwise, the client is added to the legitimate list.
  if(ShieldSys_alarm_) {
    if(!ShieldSys_isLegal(saddr)) {
      drop(p, DROP_RTR_MALICIOUS);
      return true;  //dropped, and return true.
    }
```

```
  }
  else {
    ShieldSys_push(saddr);
  }
  ShieldSys_forward(p);
  return true;
}

// Network is under DDoS attack.
void AODVUU::ShieldSys_setAlarm(void)
{
  ShieldSys_alarm_ = true;
}

// DDoS attack is over.
void AODVUU::ShieldSys_cleanAlarm(void)
{
  ShieldSys_alarm_ = false;
}

// Duplicate a packet for multiple transmission.
Packet *AODVUU::Duplicate(Packet *p)
{
  Packet *retPkt = allocpkt();
```

```
  struct hdr_ip *ih = HDR_IP(p);
  struct hdr_ip *ret_ih = HDR_IP(retPkt);
  memcpy(ret_ih, ih, sizeof(struct hdr_ip));
  return retPkt;
}

// A simple tunneling forwarding.  Only change packet head.
// A comprehensive process may include logging, return ACK,
encryption/decryption, etc.
void AODVUU::ShieldSys_forward(Packet *p)
{
  struct hdr_ip *ih = HDR_IP(p);
  rt_table_t *rt;
  struct in_addr dest_addr;
  struct in_addr saddr;
  saddr.s_addr = ih->saddr();
  static int pktcnt = 0;
  // Reset the packet header.
  ih->daddr() =  ShieldSys_providerID;
  ih->saddr() =  saddr.s_addr *1000 + pktcnt1; // Make up
an sender IP.
  ih->ttl() = 30;
  dest_addr.s_addr = ShieldSys_providerID;
  rt = rt_table_find(dest_addr);
```

```
  if (!rt || rt->state == INVALID) {
    // Route is not available, may be caused by the attack
congection.
    // Will rebuild the route first before resend the
packet.
    rreq_route_discovery(dest_addr, 0, NULL);
    //Resend packet here.
  }
  else {
    dest_addr.s_addr = rt->next_hop.s_addr;
    sendPacket(p, dest_addr, 0.0);
  }

  /* The link layer event might have changed the timer
queue,
   * so we'd better reschedule the timer queue timer...
   */
  scheduleNextEvent();
}
```

APPENDIX B

B.1 Illustration of the Experiment Result by nam Animation

The detail lay out of the top-left simulation network is illustrated in (Figure 56). The detail lay out of the top-right simulation network is illustrated in (Figure 57). The detail lay out of the bottom-left simulation network is illustrated in (Figure 58). The detail lay out of the bottom-right simulation network is illustrated in (Figure 59).

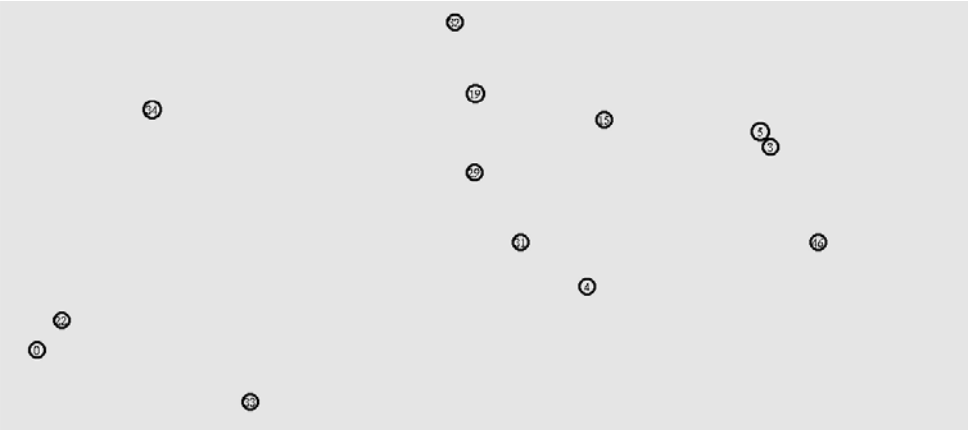


Figure 56. Top-Left Lay Out of the Simulation Network

Figure 57. Top-Right Lay Out of the Simulation Network

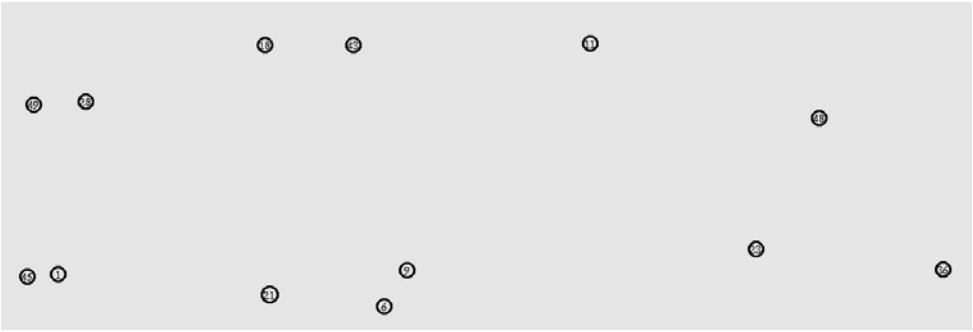


Figure 58. Bottom-Left Lay Out of the Simulation Network

Figure 59. Bottom-Right Lay Out of the Simulation Network

The nam result files are defined in the scripts, and generated by the simulation. Here are some screenshots demonstrate the playback of these nam files (Figure 60 and Figure 61).

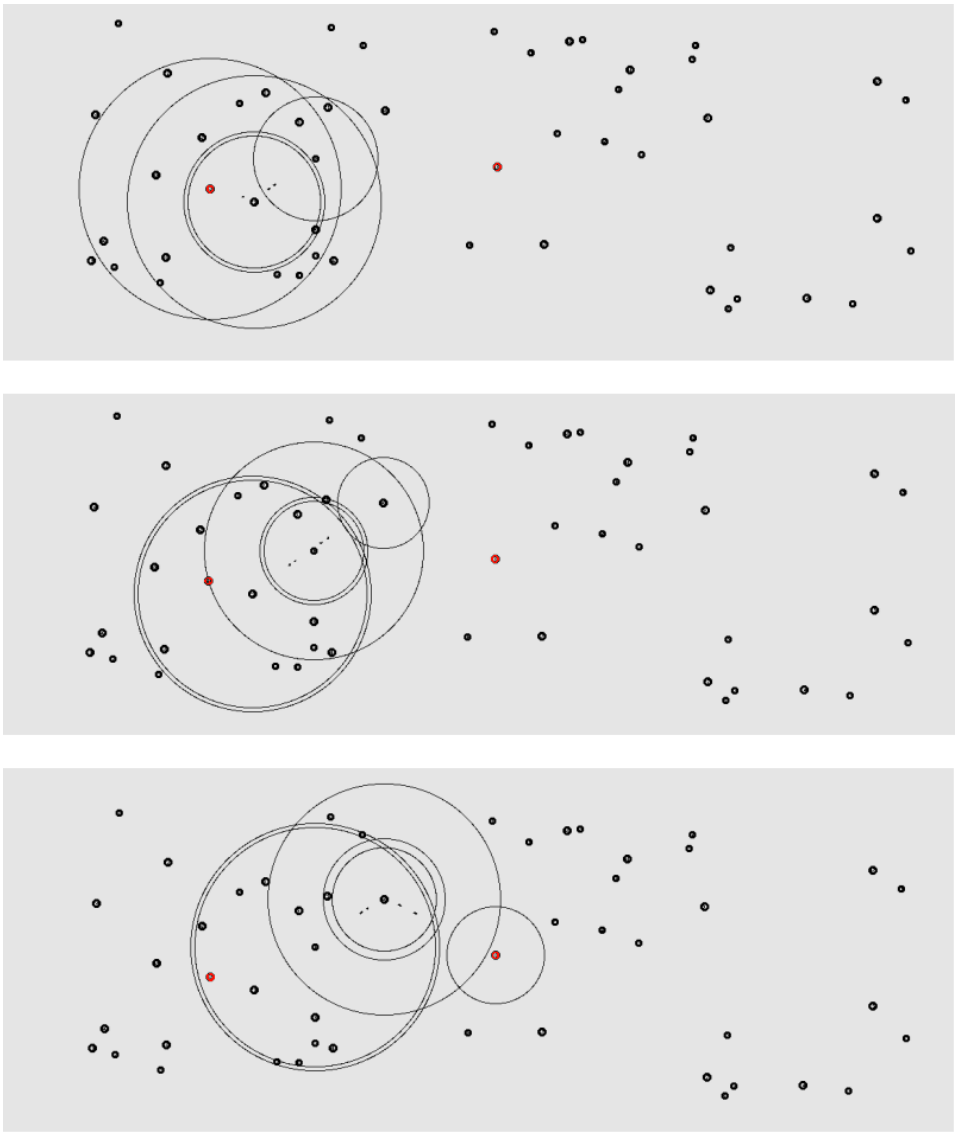


Figure 60. Screen Shots of the Path of One Data Packet Delivered From the Requester 1 to Server

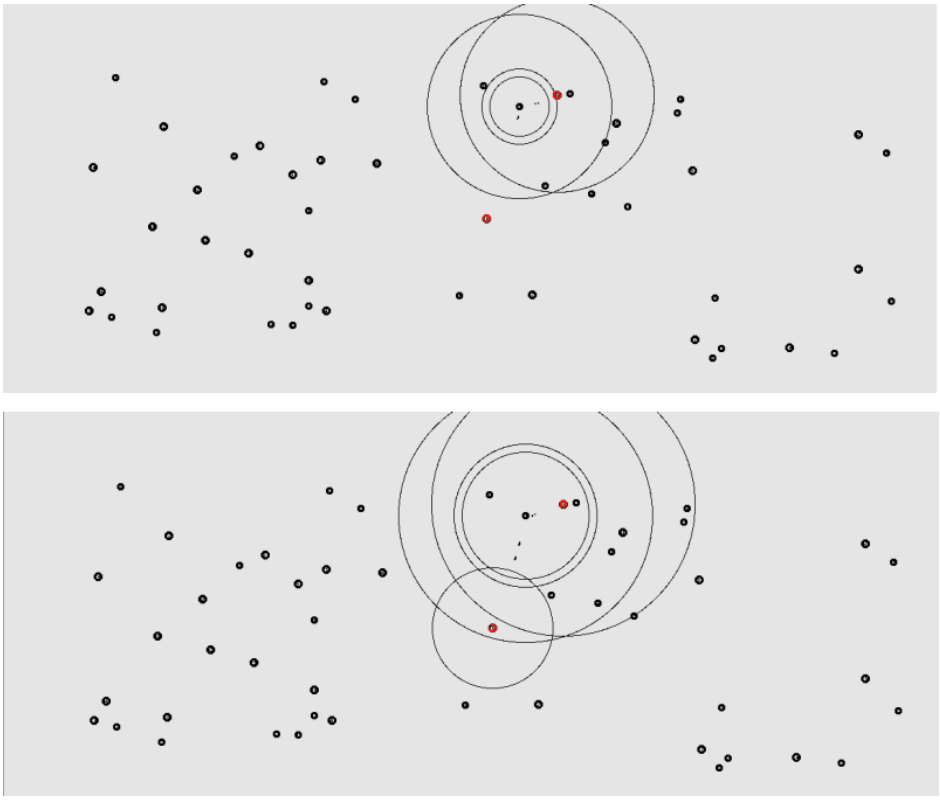


Figure 61. Screen Shots of the Path of One Data Packet Delivered From the Requester 2 to Server

The animation illustrates that the data packets are delivered from the requester to the server as expected and conforming to the AODV routing protocol.

B.2 Comparison of the Experiment Results and Model Definitions

- The delivery path of different type of packets

  A random data packet from a random service requester node 21 to the service provider node 35 in normal operation scenario:

  21 – 9 - 23 – 36 – 44 – 12 - 35

  A random attack packet from an attacker node 22 to the service provider node 35, and the path ends at the respondent guard node 31:

22 - 33 - 31

A random data packet from a random legitimate requester node 21 to the service provider node 35 in data flooding attack scenario:

21 – 9 - 23 – 36 – 44 – 12 - 35

A random data packet from a random illegitimate requester node 4 to the service provider node 35 in data flooding attack scenario, and the path ends at the respondent guard node 31:

4- 31

The verification illustrates these random packets are delivered from the sender to the receiver as expected and conforming to the AODV routing protocol.

- Table 29. The behavior of different type of nodes when the network is under a flooding attack, by average forwarding delay of 1000 packets per node of 4 nodes per type (in millisecond).

Node Type	Attack	RREQ	RREP	Legitimate Data	Illegitimate Data
Attacker	-	30.30	30.14	30.38	30.38
Node Before Guard	29.96	28.85	29.42	29.97	29.96
Node After Guard	-	28.82	28.80	29.87	-
Guard	-	30.40	30.22	30.40	-
Provider	29.98	29.01	29.55	29.97	29.97

The forwarding delays of different type of packets of different types of nodes are statistically indistinguishable. This result verifies that the network behavior and functionality of the test-bed and the system implementation are correct.

APPENDIX C

C.1 AODV Route Discovery Attack

It is not straightforward for a penetration attacker to reveal an AODV route, because each packet carries only IP addresses of two ends and the next hop. Therefore, the attacker has to keep roving around and sending routing queries to reveal the route hop-by-hop. I provide one route-tracing algorithm in Section C.2. The simulating model is described and implemented in Section C.3. The discussion is in Section C.4.

C.2 Proposed Route Tracing Algorithm

If the attacker falls into the region of a router (base node) who has the route to the victim, it roves inside the region to discover the next hop (target node), meanwhile stays inside the base region of base. The attacker may start by any position inside the region and toward any direction. The rove pattern is to move a distance (STEP) along the direction, check the signal and route information, adjust the direction if it is necessary for the next STEP, and repeat the whole procedure until the target is discovered. When the attacker loses the connection to the base, which may be caused by the out-of-region position, the attacker need roll back the latest STEP, turn clockwise by 90 degrees to try the right side, and continue the procedure. If the second STEP leads out-of-region, the

attack need roll back the latest STEP, turn clockwise by 180 degrees to try the left side, and continue the procedure (Figure 62).

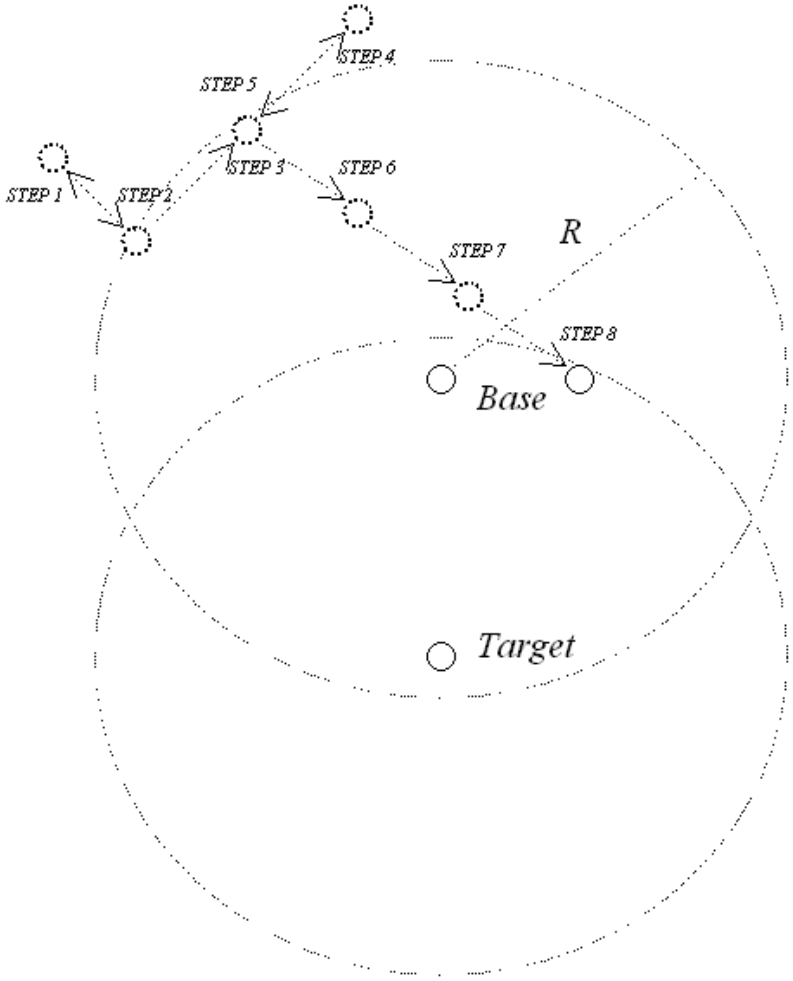


Figure 62. Illustration of a Penetration Attacker from an Arbitrary Start Point on Arc of Base and with an Arbitrary Start Direction and Discover the Next Hop of the Route, Target

To avoid the dangling dead-loop, if the attacker tried all three directions, it will roll back one more STEP, and continue to try right and left directions for the Target, and so forth (Figure 63).

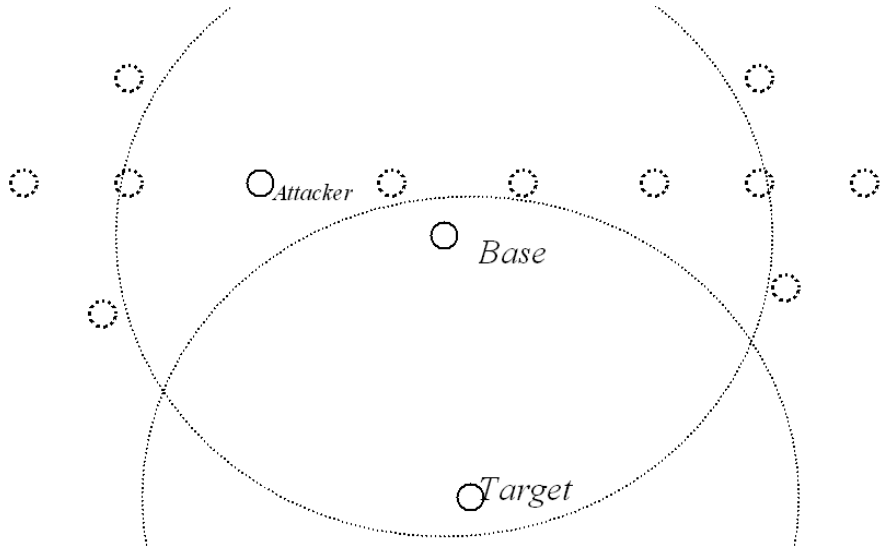


Figure 63. Illustration a Possible Dangling Dead-loop Tracing, Attacker Need More Roll Back, and Try Both Right And Left Directions Until It Reaches the Target Region

C.3 Mathematical Modeling and Implementation

For the reasons of recursion and simplification, the calculation of each hop is to find out the average number of STEP the attacker needs from the arc of the base region to the arc of the target region. The moving distance of each STEP is set to 50 meters. The moving speed of the attacker is set to 25 kilometer per hour (or 7 meter per second), which is the average speed of a bicycle [172]. This speed is also from a reasonable tradeoff between maneuverability and rapidness. The average checking delay may involve normal transmission and computation delay, and the delay from retransmission attempts. This delay is skipped in the model to maximize the attack efficiency on the proposed defense system. The distance of two nodes is from 0 to 250 according to the specifications of the simulation networks. The radian interval of start position on the arc of the Base is 1 degree or $2\pi/360$. The start direction is divided to 360 possibilities. Both start positions and the start directions are uniformly distributed. The average number of

STEP is 10.0621, which is from the probability distribution of all possible situations (Figure 64).

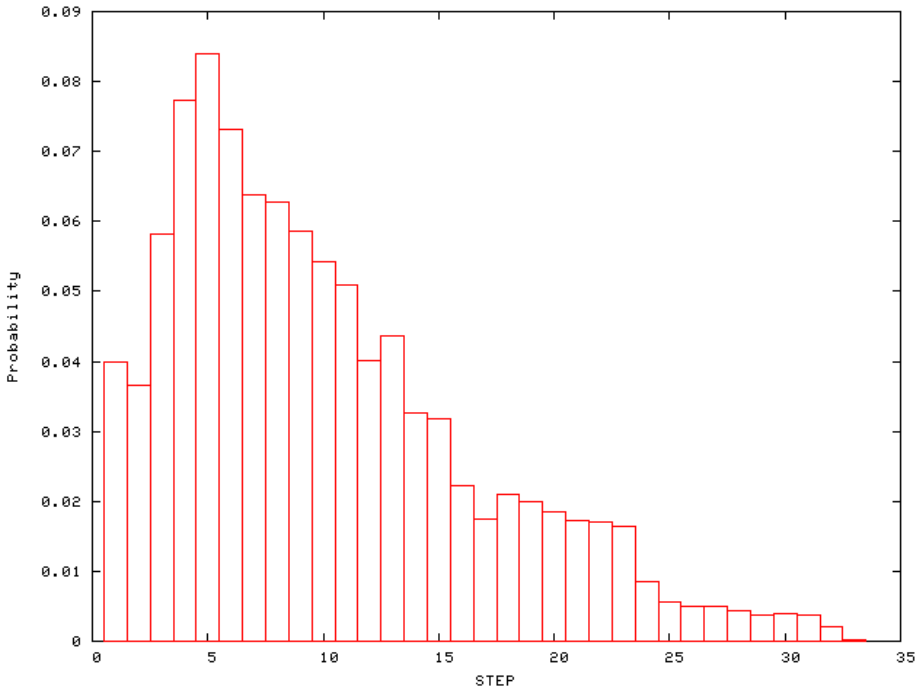


Figure 64. Probability Distribution of Each STEP number

C.4 Discussion and Defense of AODV Route Discovery Attack

According to the model, the average time to resolve one hop of a route in the simulation scenario is theoretic 10.0621 STEP × 7 second/STEP = 70.4 seconds. The hop number between a network node and a guard node is 1 to 5. Therefore, the average tracing time is 211.2 seconds, or 3.52 minutes. This is the average time an attacker may reveal the guard node, and it is the average interval a guard node needs to make a shift.

The result of the mathematical model is an ideal scenario for an attacker. But in the real world, wireless collision, node crushing, node moving, less transmission range,

and obstacle and signal shield will make the tracing much harder. And any route failure strikes out the whole tracing effort, and force the attacker start the process over again.

APPENDIX D

All nodes in this experiment are created equal. Common network nodes, DDoS attackers, defense system nodes and service provider are same network device. They have same device power, same mobility, same computing and networking capacity. The topology was generated from the CMU movement pattern tool (*indep-utils/cmu-scen-gen/setdest*) in ns-2. *setdest* needs arguments of a distribution type, number of nodes, maximum speed and simulation time. The distribution type was set to uniform. The number of nodes is 50. The maximum speed is 50 meters per second. The simulation time is 900 seconds. *setdest* randomly generates destination coordinates and moving speed for each movement according to the uniform distribution. Therefore, each node moves at a random direction with a random speed.

To especially verify the random pattern of the defense system nodes, the movement tracks of four guard nodes are individually illustrated below (Figure 65, Figure 66, Figure 67 and Figure 68). The guard node 8 started from the bottom right region; the guard node 11 started from the top right region; the guard node 29 started from the top left region; the guard node 41 started from the bottom left region. The graphics show that each guard node roamed with a fully random and long track in 900 seconds simulation.

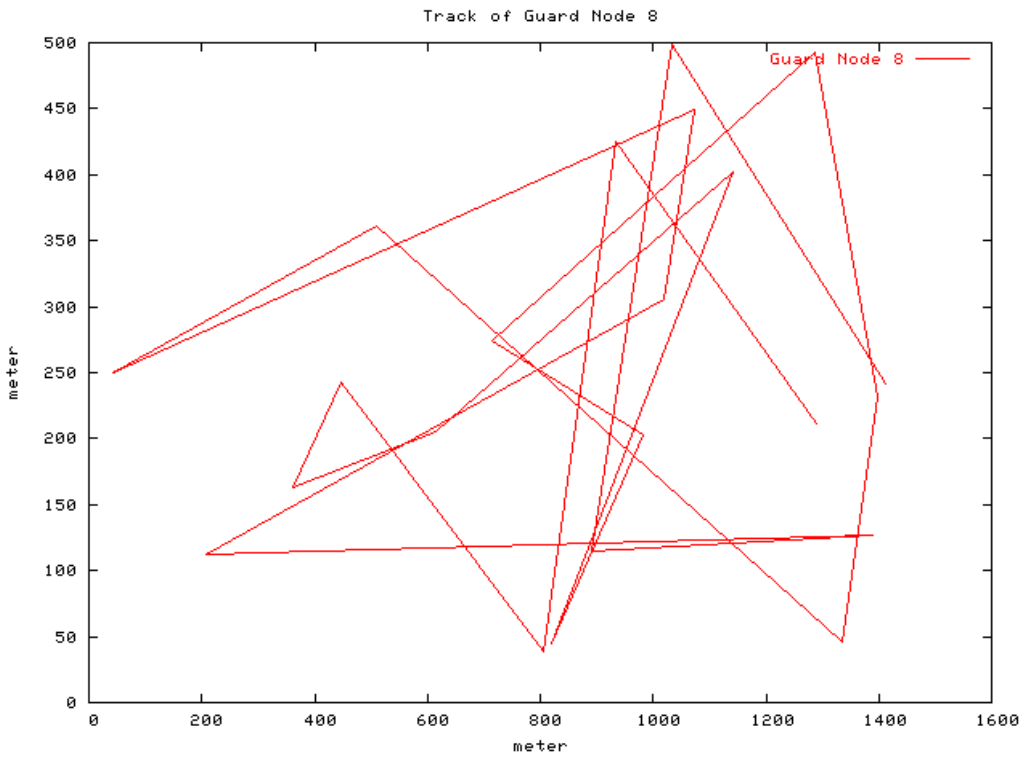


Figure 65. Illustration of Moving Track of Guard Node 8 over 900 Seconds

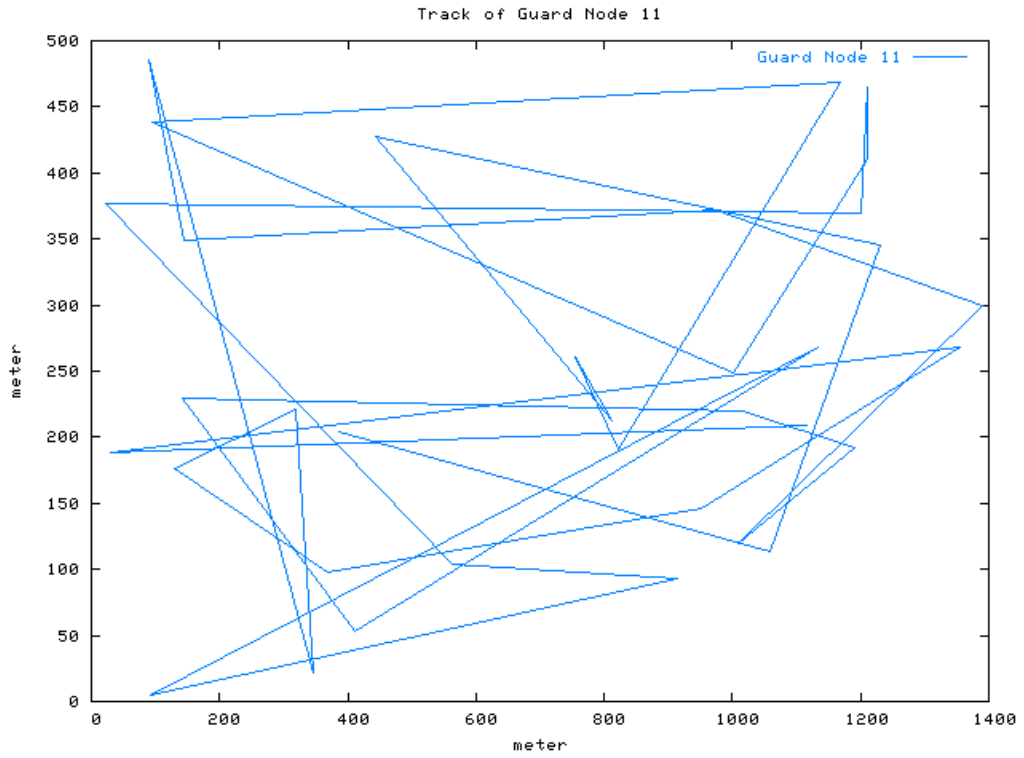


Figure 66. Illustration of Moving Track of Guard Node 11 over 900 Seconds

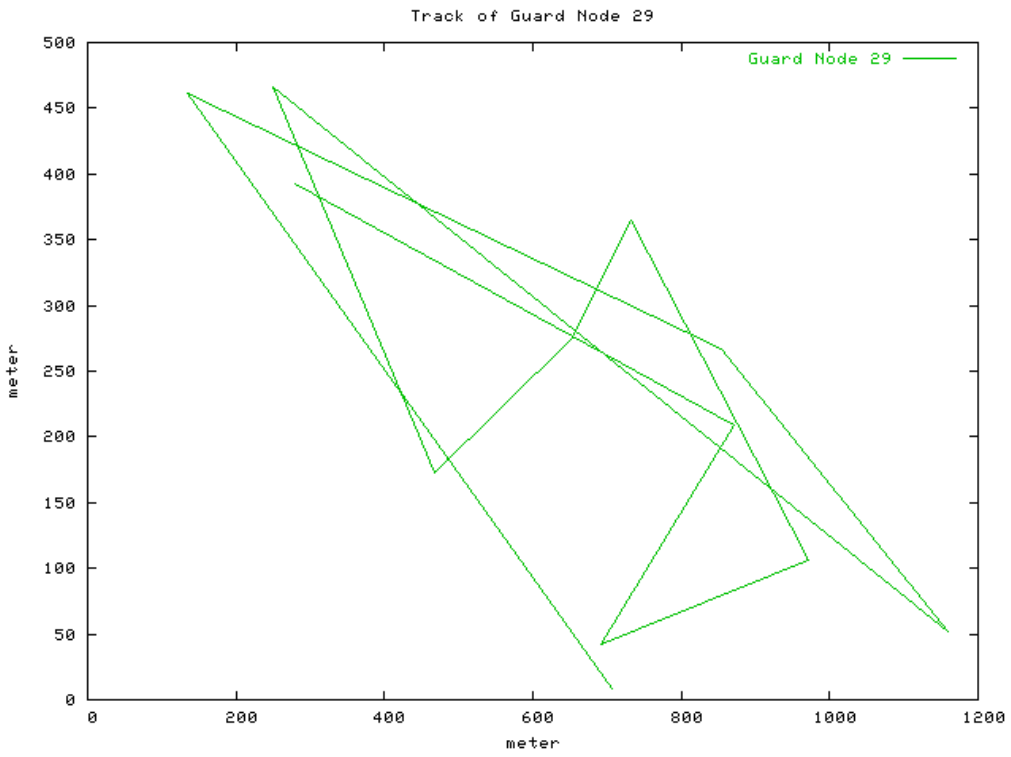


Figure 67. Illustration of Moving Track of Guard Node 29 over 900 Seconds

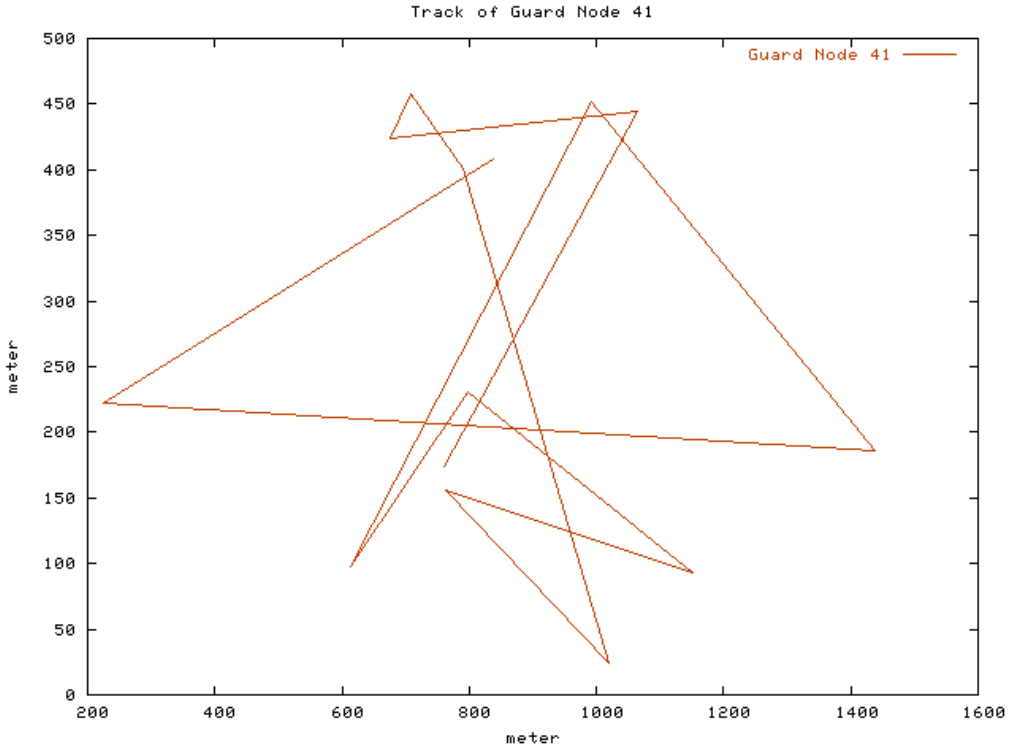


Figure 68. Illustration of Moving Track of Guard Node 41 over 900 Seconds

Destination coordinates of all guard nodes in the experiment are marked up in Figure 69. The destination coordinates are listed in Table 30. To verify the randomness of these destination locations, several tests are performed.

- Chi Square Test: $p(\chi^2) = 0.96$, it passes the test for randomness.
- Kolmogorov-Smirnov Test: $D_n = 0.09$, $C_{1-\delta} = 1.22$, $f(D_n) = 0.56 < C_{1-\delta}$, it passes the test for randomness.
- Runs Test: $z=-0.970$, $Z=1.35$, it passes the test for randomness.
- Gap Test: $p = 0.95$, it passes the test for randomness.

The test data and *setdest* passed the randomness verification.

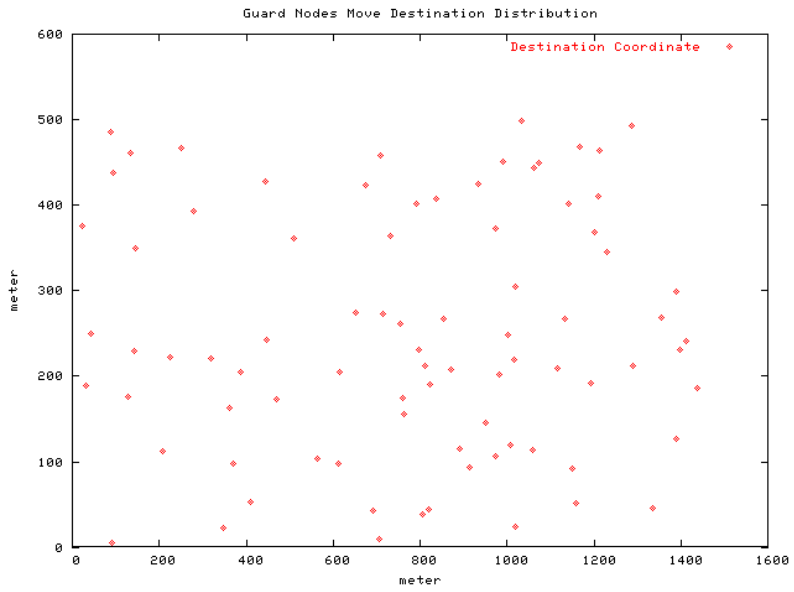


Figure 69. Movement Destination Coordinates of All Guard Nodes over 900 Seconds

Table 30. List of destination coordinates.

(387.131923, 204.342838) (1059.312257, 113.415348) (1229.716132, 345.264580)

(444.526258, 427.382998) (811.150138, 212.351451) (755.497793, 261.260566)

(823.005607, 191.088819) (1168.439624, 468.830745) (94.194177, 438.309078)
(1002.224956, 248.292377) (1210.806654, 410.998325) (1211.513834, 464.689891)
(1201.744090, 369.062921) (22.922163, 376.299347) (563.592793, 103.741366)
(915.298606, 93.297361) (91.032954, 5.360749) (1132.621527, 267.724580)
(410.957456, 53.630559) (142.503629, 229.290410) (1015.758233, 219.883900)
(1191.802087, 191.854933) (1008.883611, 120.135677) (1389.721309, 299.031015)
(973.606156, 372.504603) (144.344137, 349.237599) (88.646831, 485.739817)
(346.179385, 22.304609) (317.834007, 221.146425) (129.099859, 176.457568)
(369.190200, 97.834839) (950.640655, 146.227679) (1355.483843, 268.156777)
(30.918086, 188.735835) (1114.818955, 209.331918) (279.321380, 392.493551)
(870.348341, 208.514692) (690.767914, 42.717868) (972.316043, 106.162197)
(732.400080, 364.275673) (652.182883, 274.212766) (468.586174, 172.753940)
(249.443263, 466.700780) (1159.210332, 52.253423) (855.104878, 267.067424)
(134.959703, 461.423082) (707.319067, 9.011259) (837.612881, 407.791170)
(224.549072, 222.370744) (1437.676005, 185.851006) (991.622610, 451.164869)
(613.501244, 98.150806) (797.828642, 230.610557) (1151.546758, 92.754742)
(763.734537, 156.192890) (1018.841783, 23.916752) (791.269973, 401.375112)
(708.083234, 458.044570) (674.770609, 423.692942) (1063.283767, 444.421579)
(759.279005, 174.321066) (1289.591546, 211.828368) (934.334825, 425.391400)
(805.433520, 38.526571) (445.754587, 242.658006) (360.152093, 162.461966)
(616.116831, 204.928481) (1142.167301, 401.947883) (821.299803, 45.034776)
(981.686499, 202.542681) (714.432566, 273.675218) (1288.232717, 492.288735)
(1398.021693, 231.479202) (1334.551285, 45.725103) (510.502798, 361.396523)

(43.500882, 249.401352) (1073.795140, 448.910156) (1019.075227, 305.010843) (207.972045, 112.004191) (1389.288803, 127.087462) (891.538057, 115.190227) (1032.184539, 499.284988) (1413.009015, 241.445791)