Pulsing RoQ DDoS Attack and Defense Scheme in Mobile Ad Hoc Networks*

Wei Ren[1,2,3], Dit-Yan Yeung[2], Hai Jin[1], and Mei Yang[3]
(Corresponding author: Wei Ren)

School of Computer Science and Technology, Huazhong University of Science and Technology[1]
Luoyu Rd. 1037, Wuhan 430074, P.R. China (E-mail: renw@cs.ust.hk)
Department of Computer Science, Hong Kong University of Science and Technology[2]
Hong Kong, P.R. China
Department of Electrical and Computer Engineering, University of Nevada Las Vegas, USA[3]

(Received Nov. 8, 2005; revised and accepted Dec. 31, 2005)

Abstract

Reduction of Quality (RoQ) attack is a new style of Distributed Denial of Service (DDoS) attack. The goodput and delay performance of TCP or UDP flows are very sensitive to such RoQ attacks. In this paper, we study in detail congestion-based RoQ DDoS attacks in mobile ad-hoc networks for the first time. Specifically, we study the attacking principles based on analysis of the network capacity and classify these attacks into four categories: pulsing attack, round robin attack, self-whisper attack, and flooding attack. We then propose a defense scheme that includes both the detection and response mechanisms. The detection signals include the frequency of receiving RTS/CTS packets, frequency of sensing a busy channel (signal interference), and number of RTS/DATA retransmissions. The response scheme is based on the ECN marking mechanism. Through extensive ns2 network simulations, we demonstrate the existence of high goodput and delay jitters under the pulsing attack mode. Increase in delay (by 110 times under five attacking flows) and decrease in goodput (to 77% under five attacking flows) can be observed especially when more attacking flows occurs. Moreover, we show through simulations that similar behaviors can also be observed for TCP flows as well as networks of other topology types.

Keywords: Distributed denial of service, mobile ad-hoc network, network security, reduction of quality attack

1 Introduction

A mobile ad-hoc network (MANET) is a dynamic, self-configuring network of mobile routers and associated hosts connected by wireless links. As the routing protocols of IEEE 802.11 [5] become mature leading to increase dramatically in deployment, we expect to witness the blooming of multimedia applications on MANETs in the near future. This trend calls for higher security and quality of service (QoS) guarantees for delivering such applications reliably on MANETs.

In IEEE 802.11, the Media Access Control (MAC) mechanism depends on distributed and coordinated access of the shared transport channel. A node starts the transmission after the channel is sensed to be idle. Otherwise it begins the backoff counting phase to wait for the next channel access. The MAC protocol provides reasonable competition for and hence fair sharing of the channel. However, even in a trusted environment, the misbehaviors of some nodes that is intentional or unintentional, may lead to extremely unfair bandwidth or channel allocation. In the worst case, some nodes will essentially enter into a Denial of Service (DoS) status and can no longer function properly.

In this paper, we study congestion-based RoQ DDoS attacks in MANETs. To the best of our knowledge, our paper is the first attempt to address this problem. We first study the attacking principles based on analysis of the network capacity. We then classify the RoQ DDoS attacks into four categories, namely, pulsing attack, round robin attack, self-whisper attack, and flooding attack. To counter these attacks, we propose a defense scheme that includes both the detection and response mechanisms. Detection relies on three signals from the MAC layer. Simulation results obtained using the ns2 network simulator demonstrate the existence of large jitters in the goodput and delay, which are harmful to both bandwidth-aware applications (e.g., video-based multimedia) and delay-aware applications (e.g., voice-based multimedia).

The rest of this paper is organized as follows. We dis-

*National Natural Science Foundation of China (Project No. 90412010), National Basic Research Program (973) of China (No. 2003CB317003), Research Grants Council of Hong Kong SAR (Project No. AoE/E-01/99).

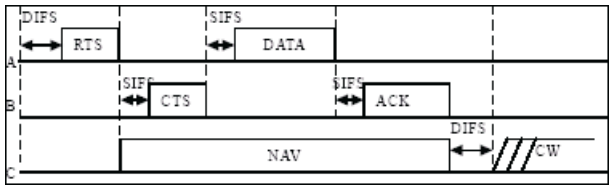

Figure 1: 802.11 DCF packet exchange

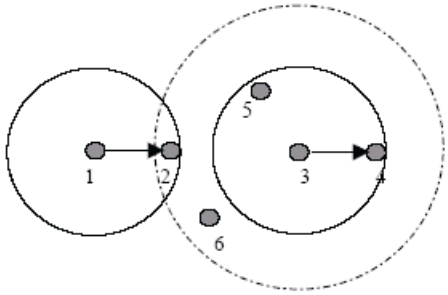

Figure 2: Basic idea of channel competition

cuss the attacking principles, attack categories and defense scheme in Section 2. In Section 3, we present extensive simulation results to demonstrate the performance degradation caused by the attacks. We discuss some related work in Section 4 and then conclude the paper in Section 5.

2 DDoS Attacking Patterns and Defense Scheme

2.1 Overview of IEEE 802.11 DCF

The Distributed Coordination Function (DCF) of IEEE 802.11 specifies the use of CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance) to reduce packet collisions in a network. A node with a packet to transmit picks a backoff value I which is chosen uniformly randomly from the set $\{0, 1, \ldots CW\}$ where CW denotes the size of the contention window, and then transmits the packet after waiting for I idle time slots. Nodes exchange Request to Send (RTS) and Clear to Send (CTS) packets to reserve the channel before transmission. Both RTS and CTS packets contain a duration field to indicate the time required for utilizing the channel to complete the data transmission. Other hosts that overhear either the RTS or the CTS are required to adjust their Network Allocation Vector (NAV), which specifies for how long the node should defer transmissions on the channel. If a transmission is unsuccessful (by the lack of CTS for RTS or ACK for the DATA sent), the value CW of is doubled and the lost packet is retransmitted. The maximum number of retransmissions of RTS is always set to 7 and that of DATA to 4. On the other hand, if the transmission is successful, the host resets its CW to a minimum value CW_{min}, There are additional idle times between frames, such as DIFS (Distributed Inter Frame Space) and SIFS (Short Inter Fame Space). Figure 1 depicts the procedure of packet exchange between a sender (A) and a receiver (B) and the behavior of a listener (C).

2.2 Attacking Principles

In wireless networks, the main competing resource is the channel, which is shared by wireless nodes with only one node having access at a time. Unlike wired networks in which channel congestion is always the result of increased rate of competing flows at the bottleneck link, congestion in MANETs may also be due to the aggregation of mobile nodes. If the attacking nodes aggregate with high density near the victim nodes, the attacking nodes may occupy the channel in the most of the time. Another reason of potential attacking is the current routing protocols or intermediate nodes in MANET have not provided traffics control mechanisms, such as traffics filter, traffics allocation and Quality of Services.

According to the analysis by Gupta and Kumar [3], radios that are sufficiently distant from each other can transmit data concurrently. The total number of data packets that can be simultaneously transmitted for one hop increases linearly with the total area of the ad-hoc network. If the node density is constant, then the total one-hop capacity is $O(n)$ where n denotes the total number of nodes in the network. As the network grows in size, the number of hops between the source and destination nodes may also increase. The average path length grows with the spatial diameter of the network, which is proportional to the square root of the area, i.e., $O(\sqrt{n})$. Therefore, the total end-to-end capacity is roughly $O(n/\sqrt{n})$ and the end-to-end throughput available to each node is $O(1/\sqrt{n})$. To simplify the analysis, given n nodes in the interference range, each node has a probability of $1/n$ to access the channel. Of the n nodes there are m attacking nodes, so each victim node only has a probability of $(1 - m/n)$ to access the channel.

Figure 2 illustrates the basic idea of channel competition. We use a solid-line circle to indicate the transmission range (250m) and a slash-dot-line circle to indicate the interference range or sensing range (550m). Attacking node 3 sends a packet to node 4. Node 5 is within the transmission range of node 3 and node 2 and node 6 are within its interference range. So these three nodes have to wait in order to access the channel to communicate with node 3. While node 5 will receive RTS/CTS packets, node 2 and node 6 cannot receive them. However, they can sense the busy channel and persist within the backoff stage. If other nodes send packets to these nodes, e.g., node 1 sends a packet to node 2, node 2 will have no response because of the interference from node 3.

2.3 Four Attack Types

Based on the characteristics of MANETs and the MAC protocol, we describe in this subsection different DDoS attack patterns and hence different attack types. In our discussions, we adopt the basic assumption that the general requirements for information security in MANETs have been satisfied, such as through message encryption and node authentication. Instead, our focus is on the network security aspect of MANETs. We are particularly interested in attack patterns that are sophisticated attacks designed by expert adversary and cannot be detected easily. Actually many other simple attacks are included in our modelled attacking patterns as follows.

Let the set of all nodes in a MANET be S : $\{N_1, N_2, \ldots N_n\}$, the set of attacking nodes be S_a : $\{N_{a1}, N_{a2}, \ldots N_{an}\}$ with $S_a \subset S$, and the set of victim nodes be S_v : $\{N_{v1}, N_{v2}, \ldots N_{vn}\}$ with $S_v \subseteq S - S_a$. In the context of DDoS, the attacking nodes include the active attacking nodes and the passive zombie nodes that are compromised by the active attacking nodes or are controlled by them to become their slaves. Here we simply refer to all of them as attacking nodes.

The attacking nodes cause chaos in the channel by sending packets arbitrarily. To prevent them from being detected, they continuously change the packet size and time as well as the sending and receiving nodes. In essence, it is a channel-consuming attack in which the attacking nodes compete for the channel aggressively and occupy it for a long time. As a consequence, the channel is always in a saturated state and hence the victim nodes essentially enter into a DoS status. The purpose of this attack is to consume network bandwidth and produce traffic overhead in a smart way using low attacking cost. The direct outcome of such attack is the reduction of quality of service of the channel and localized congestion near the victim nodes. In what follows, we discuss four different attack types based on the different attacking patterns. Figure 3 shows the four types of attacking patterns.

Pattern 1: Pulsing Attack:
A single attacking node $N_{ai} \in S_a$ sends packets to a randomly selected victim node $N_{vi} \in S_v$, with a random sending period T and a random packet size P_i.

Pattern 2: Round Robin Attack:
Multiple randomly selected attacking nodes $N_{ai1}, N_{ai2} \ldots N_{ain} \in S_a$ send packets in sequence in a round robin manner to randomly selected victim nodes $N_{vj1}, N_{vj2} \ldots N_{vjn} \in S_v$, with a random sending period T_i and a random packet size P_i.

Pattern 3: Self-Whisper Attack:
Two randomly selected nodes N_{a_p}, N_{a_q} in S_a send packets to each other with a random sending period T_i and a random packet size P_i.

Pattern 4: Flooding Attack:
Multiple randomly selected attacking nodes send packets to a single victim node with a random period T_i and a random packet size P_i. The purpose of the attack is to force the victim node to decrease its communication with other nodes and eventually enter into a DoS status.

2.4 Discussions

We discuss here some protocol attacking patterns that potentially are alternative ways for launching DDoS attacks. However, these attacking patterns have to modify the MAC protocol in order to occupy the channel greedily. This type of attack depends on the implementation of the MAC protocol because some implementations are hard-coded in firmware. Possible attacking patterns include using a small CW value, fixing the CW value in the retransmission backoff stage without doubling, spoofing the NAV value, forging the RTS/CTS packets, dropping the RTS/DATA packets, forging the routing protocol, etc. However, these attacking patterns may not be possible in some stack implementations since the MAC layer is hard-coded in firmware.

While such attacking patterns involve forged protocol packets violating protocol specifications that can be detected easily, the attacking patterns studied in our work can be implemented easily without modifying the protocol stack and yet they cannot be detected easily. Therefore, it suffices to control the nodes to send packets only, regardless of whether they are TCP or UDP based. Moreover, the randomness involved in choosing the attacking nodes, the sending period, the packet size and the packet type makes it more difficult for the victim nodes to detect the attacks effectively.

It is worth noting that the four attacking patterns described above are not totally independent of each other. Also, some attacking patterns may be combined together. Moreover, although some victim nodes are not the direct targets of the attacks, their performance may also be affected indirectly.

2.5 Our Defense Scheme

To defend against such DDoS attacks, we propose a defense scheme that includes the detection and response stages.

Detection makes use of three status values that can be obtained from the MAC layer: frequency of receiving RTS/CTS packets, frequency of sensing a busy channel, and the number of RTS/DATA retransmissions. When the number of RTS/CTS packets received exceeds a certain threshold RTS/CTS_{thresh}, it indicates that too many nodes are within the transmission range to compete for the channel. When the channel is sensed to be in a busy state, a node will persist in the backoff stage and stop the CW count. When the stopping time is longer than a threshold Sensing_{thresh}, it indicates that too many nodes are within the interference range. In general, if the num-

ber of retransmissions for RTS packets is larger than 7 and that for DATA packets is larger than 4, the packets will be dropped. Thus if the number of retransmissions exceeds a threshold RET_{thresh}, it will be regarded as an indicator for channel congestion. Since these status values are already available in the protocol stack implementation, the overhead required for implementing this detection scheme is very low.

During the response phase, the nodes will mark each packet with an Explicit Congestion Notification (ECN) bit to notify the sender nodes and keep a list of these nodes. The sender nodes, upon seeing these packets with ECN marking, will then reduce their sending rate. If the channel continues to be congested because some sender nodes do not reduce their sending rate, these nodes will be considered as attacking nodes. ECN marking may be integrated into lower protocol packets, such as the routing protocol or MAC protocol, if the transport protocol is only in a single direction. For sender nodes that are cooperative in reducing their sending rate, they are still recorded in the list of nodes that also includes the (non-cooperative) attacking nodes. The behaviors of all the nodes in the list will be analyzed to build a set of nodes N_{v_i}.

3 Simulation Experiments

3.1 Simulation Setup

We use the network simulator ns2.26 [11] and the wireless extension from the CMU Monarch Project [10] for our simulation experiments. The parameter settings for the simulations are: the radio propagation mode is TwoRay-Ground, antenna type is omni antenna, interface queue length is 50 (packets), queue management scheme is Drop-Tail, routing protocol is AODV, height of antenna is 1.5m, transmission distance is 250m, signal interference or sensing distance is 550m, and signal transmission rate is 2M. Other simulation parameters are given in Table 1.

There are totally 36 nodes in the simulated network covering a simulated area of 1500m x 1500m. The distance between nodes is 100m. We simulate the static grid scenario. Figure 4 depicts the simulation topology and the victim and attacking flows. The UDP-based victim flow from node 14 to node 17 simulates voice or video traffic at a rate of 0.3Mb. The attacking flows, also UDP based, occur in the neighborhood of node 14 from node 20 to 26, 19 to 18, 13 to 12, 7 to 6, and 8 to 2 in the pulsing mode. UDP packets are sent every 5s and persist for 1s at a rate of 0.3Mb. While the victim flow occurs from 2s to 280s, the attacking flows are from 40s to 240s.

We use two measures for performance evaluation of the victim flow in our experiments:

1) End-to-end delay: time taken by a packet sent by the sender to arrive at the receiver.

Table 1: Simulation parameter values of PHY, MAC and UDP

Parameter	Our First (Second)
Time slot	20us
DIFS	50us
SIFS	10us
RTS length	160bits
CWmin	31
CWmax	1023
UDP packet length	512B

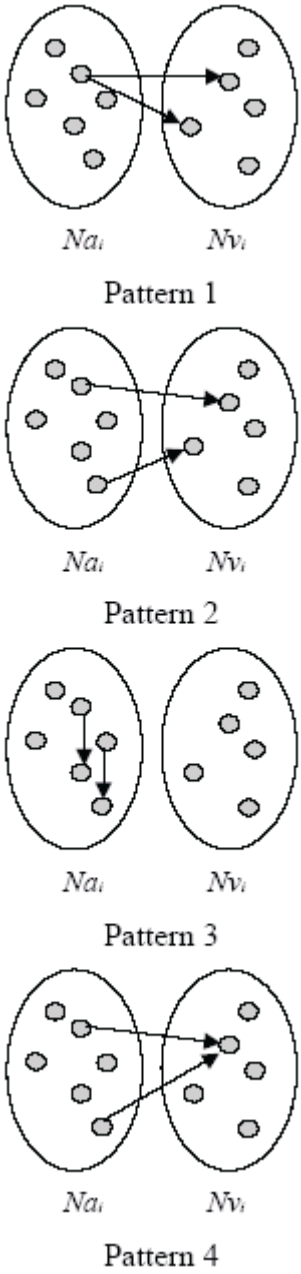

Figure 3: Four different attacking patterns

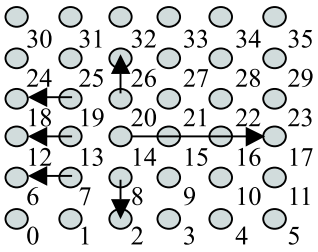

Figure 4: Simulation topology with victim and attacking flows

2) Goodput: volume of data successfully received by the receiver per second.

3.2 Simulation Results

In the first set of simulation experiments, a new attacking flow is added every 40s and all attacking flows stop at 240s. Figure 5 depicts the simulation results, showing the goodput, delay, interface queue (IFQ) drop due to overflow, and router overhead as a function of the simulation time. We can see that the goodput of the victim flow decreases with the addition of attacking flows. The jitters observed are due to the pulsing mode of the attack, which persists for 1s and then becomes idle for 4s in each period of 5s. Such jitters are harmful to goodput-sensitive multimedia applications such as video-based applications. With the addition of attacking flows, the delay of the victim flow also increases with jitters that are harmful to delay-sensitive multimedia applications such as voice-based applications. Since the senders of all the attacking flows are near the sender of the victim flow, they compete for the channel and hence IFQ drop corresponds to channel congestion. The drop also increases with the addition of attacking flows. We can see that the router overhead increases by 0.3Mb every 40s with the addition of a new attacking flow, showing that the increase in AODV packets is almost equal to the traffic volume of the additional attacking flow.

In the second set of simulation experiments, we report the average goodput and average delay performance under different numbers of attacking flows. Unlike the first set of simulations that increases the number of attacking flows gradually, here the same number of attacking flows, from 0 to 5, is used in each simulation run throughout the entire simulation duration from 40s to 240s. Table 2 gives more details of the results. With five attacking flows, the average goodput decreases to 77% of that without attack while the average delay increases by almost 110 times.

3.3 Discussions

In the simulation experiments discussed above, UDP flows are simulated for both victim and attacking flows. This is typically found in multimedia applications. In fact, either or both may also be TCP based. We have also conducted

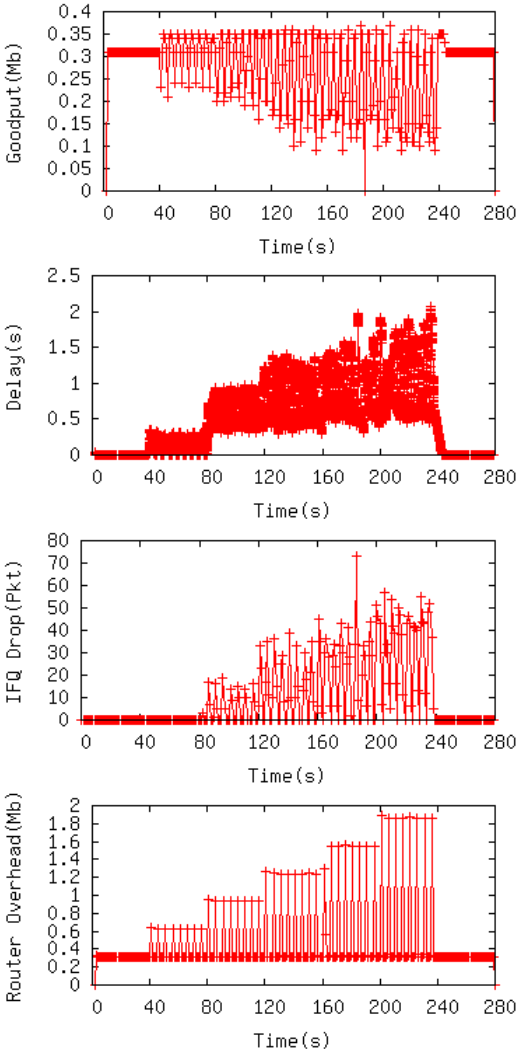

Figure 5: Results from first set of UDP-based simulation experiments

Table 2: Average goodput and average delay of victim flow under different numbers of attacking flows

Attacking flows	Average goodput (Mbps)	Ratio	Average delay (ms)	Ratio
0	0.31	1	5.77	1
1	0.31	1	84.92	14.71
2	0.30	0.97	405.24	70.22
3	0.27	0.87	470.27	81.49
4	0.26	0.84	553.81	95.96
5	0.24	0.77	633.17	109.72

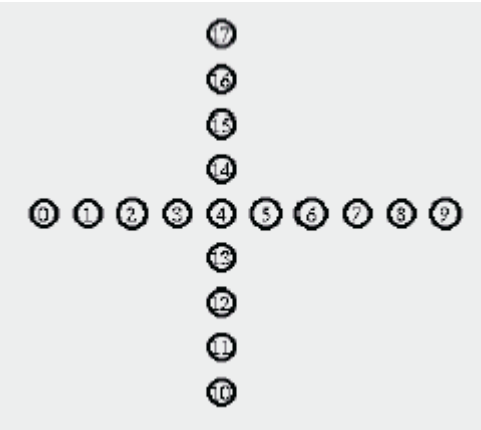

Figure 6: Cross topology

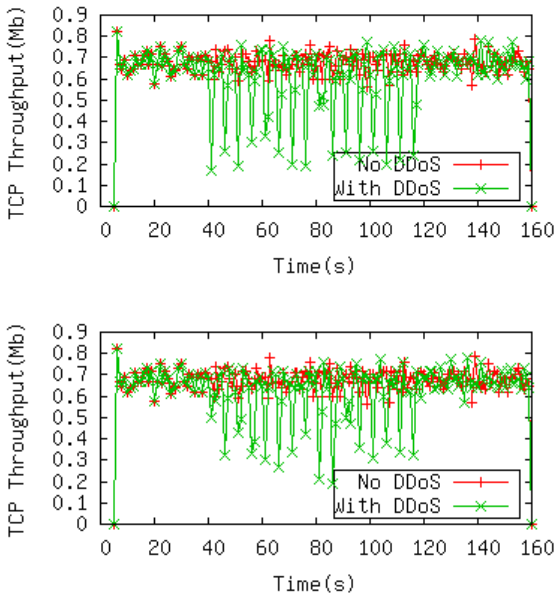

Figure 7: Cross topology

simulation experiments by using TCP flows or UDP flows to attack a TCP flow. The results are almost the same as those reported above.

Besides the grid network topology, we have also tried other topologies, including chain and cross- topology giving similar findings. Due to space limitation, we will only provide some results for the cross topology here.

Figure 6 depicts the cross network topology used in our simulations. The distance between nodes is 50m. The TCP-based victim flow is from node 2 to node 7, while the attacking flow that is either TCP or UDP based is from node 11 to node 16. We use NewReno TCP in our simulations. The UDP rate is 0.3Mb. The TCP victim flow is from 5s to 160s while the TCP or UDP attacking flow is from 40s to 120s. The attack uses the same pulsing pattern as before, which sends packets that persist for 1s and then remain idle for 4s in each period of 5s.

Figure 7 shows the throughput of the TCP victim flow with and without attack. The upper graph corresponds to a UDP-based attack while the lower graph corresponds to a TCP-based attack. As expected, for both cases, the throughput drops with jitters as soon as the pulsing attack starts.

4 Related Work

A survey of security issues in MANETs can be found in [12]. However, its focus is on secure routing protocols and key management schemes rather than MAC DDoS attacking behaviors. Zhang and Lee [13] discuss different possible attacks to the different layers in the protocol stack. However, they leave detailed descriptions of the attacking patterns to the future work. Gupta et al. [4] point out possible attacks to the routing layer and MAC layer even in networks with end-to-end authentication. If there are two collusion nodes, in which one is a sender and the other is a receiver, they can launch such attacks easily. However, the attacking patterns used in their simulation study are very simple and are easy to detect. Moreover, they do not consider the possibility of distributed attacks.

Kyasanur and Vaidya [7] propose to modify the IEEE 802.11 MAC protocol to solve the misbehavior problem of selfish nodes. In their scheme, the receiver will determine the backoff value of the sender, so the receiver can punish the sender by increasing the backoff value when the sender is found to misbehave. Noubir and Lin [9] describe a type of DoS attack that can lead to high power consumption for the victims. They argue that the NAV value is vulnerable in the RTS/CTS handshake packets, since the attacker can utilize this value to estimate the transmission event and then send data to interfere the normal frame. This will result in frequent retransmission of the normal frame, causing high power consumption on the normal side while keeping that on the attacking side low. However, this type of attack can be detected easily due to its obvious conflict with the sending behavior defined in the MAC protocol. MacKenzie and Wicker [8] use game theory to deal with the problem of selfish nodes. They design a distributed protocol to make the nodes converge to the Nash equilibrium of the bandwidth by allocating a certain cost to each node before it accesses the channel.

Aad et al. [1] analyze DoS attacks to closed-loop protocols such as TCP and open-loop protocols such as UDP. They describe a JellyFish attack to TCP through packet disordering, periodic packet dropping and delay variance jittering to cause maladjustment of the TCP functions, such as RTT measurement, RTO estimation, slow start, congestion avoidance, etc. They also describe a Black Hole attack to UDP, in which the nodes along the path drop packets like a black hole. However, such attacking behaviors require modification of the normal packet forwarding mechanism.

For DDoS attacks on wired networks, [6] and [2] de-

scribe shrew attacks and RoQ attacks. Shrew attacks can make TCP go into a timeout status and enter the slow-start phase frequently. On the other hand, RoQ attacks compromise the protocol vulnerability for the reduction of quality of service. To the best of our knowledge, there has not been previous research on studying RoQ attacks in wireless networks.

5 Conclusion

In this paper, we study congestion-based RoQ DDoS attacks in MANETs for the first time and propose four possible types of such attacks. We also discuss a detection scheme that monitors three MAC layer signals and a response scheme based on ECN marking.

Network simulation experiments on pulsing attacks show that great jitters in the goodput and delay performance can be observed. This results in frequent IFQ dropping and high increase in the router overhead especially when the number of attacking flows increases. With five attacking flows, the goodput of the victim flow drops to 77% while the delay increases by almost 110 times. Increase in delay and decrease in goodput can be observed especially when more attacking flows occurs. Moreover, we show through simulations that similar behaviors can also be observed for TCP flows as well as networks of other topology types.

Acknowledgements

We would like to thank Samuel Chanson, Wei Chen and Ivan Lam for insightful discussions and the anonymous reviewers for useful comments. This research has been supported by the National Natural Science Foundation of China under grant number 90412010, National Basic Research Program (973) of China No. 2003CB317003, and the Research Grants Council of the Hong Kong Special Administrative Region under the Area of Excellence (AoE) scheme (project AoE/E-01/99).

References

[1] I. Aad, J. P. Hubaux, and E. W. Knightly," Denial of service resilience in ad hoc networks," in *Proceedings of MOBICOM2004*, pp. 202-215, Philadelphia, Pennsylvania, USA, Sept. 2004.

[2] M. Guirguis, A. Bestavros, I. Matta, and Y. Zhang, "Reduction of quality (RoQ) attacks on internet end-systems," in *INFOCOM2005*, vol. 2, pp. 1362-1372, Miami, Florida, Mar. 2005.

[3] P. Gupta and P. R. Kumar, "The capacity of wireless networks," *IEEE Transaction on Information Theory*, vol. 46, no. 2, pp. 388-404, Mar. 2000.

[4] V. Gupta, S. Krishnamurthy, and M. Faloutsos, "Denial of service attacks at the MAC layer in wireless ad hoc networks," in *IEEE MILCOM*, vol. 2, pp. 1118-1123, Oct. 2002.

[5] IEEE, *IEEE Standard for Information Technology V Telecommunications and Information Exchange between Systems V Specific Requirements V Part 11: Wireless LAN MAC and PHY Specifications*, IEEE Std 802.11-1999, IEEE, New York, 1999.

[6] A. Kuzmanovic and E. W. Knightly, "Low-rate TCP-targeted denial of service attacks," in *Proceedings of the 2003 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM2003)*, pp. 75-86, Karlsruhe, Germany, Aug. 2003.

[7] P. Kyasanur and N. Vaidya, "Detection and handling of MAC layer misbehavior in wireless networks," in *Proceeding of the International Conference on Dependable Systems and Networks (DSN)*, June 2003.

[8] A. B. MacKenzie and S. B. Wicker, "Stability of multipacket slotted aloha with selfish users and perfect information," in *Proceedings of IEEE INFOCOM 2003*, vol. 3, pp. 1583-1590, San Francisco, CA, Apr. 2003.

[9] G. Noubir and G. Lin, "Low-power DoS attacks in data wireless LANs and countermeasures," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 7, no. 3, pp. 29-30, 2003.

[10] The CMU Monarch Project, *Wireless and Mobility Extension to ns.*

[11] The Network Simulator NS-2, http://www.isi.edu/nsnam/ns/index.html.

[12] H. Yang, H. Luo, F. Ye, S. Lu, and L. Zhang, "Security in mobile ad hoc networks: Challenge and solutions," *IEEE Wireless Communications*, vol. 11, no. 1, pp. 38-47, Feb. 2004.

[13] Y. Zhang and W. Lee, "Intrusion detection in wireless ad hoc networks," in *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MobiCom2000)*, pp. 275-283, 2000.

Wei Ren received his B.S. and M.S. degrees from the University of Science and Technology Beijing and Ph.D. degree from Huazhong University of Science and Technology (HUST) in 1996, 1999 and 2006 respectively. All are in Computer Science. From 2004-2005 he was a visiting scholar in the Department of Computer Science, Hong Kong University of Science and Technology (HKUST). From 2006 he has been a postdoctoral researcher in Department of Electrical and Computer Engineering, University of Nevada Las Vegas. He has published over 20 international journal and conference research papers. He also served as a reviewer for ICC and CACM and a program committee member for iiWAS2005. His recent research interests include network security, mobile ad hoc networks, wireless sensor networks and performance evaluation. Email: renw@cs.ust.hk or renw@cs.unlv.edu.

Hai Jin received his B.S., M.S., and Ph.D. degrees in Computer Science in 1988, 1991, and 1994, respectively, from the Huazhong University of Science and Technology (HUST), Wuhan, China. In 1996, he was awarded German Academic Exchange Service (DAAD) fellowship to visit the Technical University of Chemnitz in Germany. He worked at the University of Hong Kong (HKU) between 1998 and 2000. He also worked as a visiting scholar at the University of Southern California (USC), USA from 1999 to 2000. He is member of IEEE and ACM. He is the Chief Scientist of the ChinaGrid Project. He is now the Professor and the Dean of School of Computer Science and Technology of HUST. He is also the Director of the Cluster and Grid Computing Lab (Key Lab in Hubei Province). He has served on the editorial board of seven international journals and as program committee member for more than 100 international conferences and workshops. He has co-authored 7 books and published over 150 research papers. His research interests include grid computing, P2P computing, cluster computing, network storage, network security, and pervasive computing. Contact him at hjin@hust.edu.cn.

Dit-Yan Yeung received his BEng degree in electrical engineering and MPhil degree in computer science from the University of Hong Kong (HKU), and PhD degree in computer science from the University of Southern California (USC). He was an Assistant Professor at the Illinois Institute of Technology (IIT) before he joined the Hong Kong University of Science and Technology (HKUST). He is now an Associate Professor and also the Associate Head of the Department of Computer Science of HKUST. Email: dyyeung@cs.ust.hk.

Mei Yang received her PhD in computer science from University of Texas at Dallas in 2003. She is now an Assistant Professor of Department of Electrical and Computer Engineering, University of Nevada Las Vegas, USA. Email: meiyang@egr.unlv.edu.