Validating Reliability of OMNeT^{++} in Wireless Networks DoS Attacks: Simulation vs. Testbed

Mina Malekzadeh[1], Abdul Azim Abdul Ghani[2], Shamala Subramaniam[3], and Jalil Desa[4]

(Corresponding author: Mina Malekzadeh)

Faculty of Computer Science and Information Technology, University of Putra Malaysia
43300 Serdang, Selangor, Malaysia
(Email: {minarz[1], drshamala[3]}@gmail.com, azim@fsktm.upm.edu.my[2], dralil@tmrnd.com.my[4])

(Received Mar. 19, 2010; revised and accepted Apr. 3, 2010)

Abstract

Despite current 802.11i security protocol, wireless networks are vulnerable to Denial of Service (DoS) attacks. Sending a continuous stream of forgery control frames by an attacker can easily flood wireless channel so that the network cannot be available for its associated users. These attacks are possible because wireless control frames do not carry any cryptographic mechanism to detect and discard forgery frames. In this research in parallel to our experiments, we develop an extension module for wireless DoS attacks using OMNeT^{++} to assess the reliability of this simulation tool in compare to our real 802.11 wireless network testbed. To fulfill these goals, throughput, end-to-end delay, and pocket lost ratio are considered as our performance measures running on both real testbed and simulation model. The results are used as a comparative acceptance of the simulation environment. Hereby we can confirm accuracy of the simulation results and OMNeT^{++} in wireless DoS attack domain.

Keywords: DoS attack simulation, OMNeT^{++}, wireless DoS attacks, wireless security, 802.11 networks

1 Introduction

As wireless network popularity increases, the risk of security attacks increases as well. DoS attack is one of the most common types of security threats against wireless networks [1, 16]. Current security protocols in WLAN such as WEP, WPA, and 802.11i just carry cryptographic check for data frames. Therefore short and unprotected wireless control frames can easily be forged by an attacker to conduct variety types of DoS attacks against wireless networks [8, 10, 11]. An attacker can send a continuous stream of forgery control frames with large duration value to keep wireless channel busy with flooding of useless data which causes to consume limited resources of wireless network.

Due to the complexity and difficulty to design and implement real networks for experiments, simulation tools are widely employed. Simulation tools allow to investigate different scenarios and validate them before deployment which can decrease both time and cost. Objective modular network testbed in C^{++} (OMNeT^{++}) is an open source C^{++} environment with a GUI support which makes it more attractive for researches especially in communication network area.

Three main contributions of this research are described as:

- Design and develop a new extension module for OMNeT^{++}. We apply this new extension module to make OMNeT^{++} capable to simulate variety types of wireless DoS attacks using forgery control frames.
- Design a real testbed to experiment the same wireless DoS attacks like our simulation model over a real 802.11 wireless network testbed.
- Calculate the metrics as end-to-end delay, throughput, and pocket lost ratio over both testbed and simulation environment. We compare result of simulation environment with the measuring results from the testbed to evaluate accuracy of OMNeT^{++} by using the extension module in case of heavy loads of DoS attacks.

The rest of this paper is organized as follows. In Section 2 related works are deliberated. DoS attacks using wireless control frames are discussed in Section 3. In Section 4 overview of OMNeT^{++} is done. Sections 5 and 6 present topology of our simulation model and the simulation results respectively. Sections 7 and 8 illustrate structure of our testbed and experimental results respectively. Section 9 concludes our work.

2 Related Works

Security issue related to wireless networks has been paid attention by a lot of wireless network researches. In [11] they consider RTS, CTS, and ACK which are used by attacker to lunch DoS attacks over WLAN. In a testbed they present effectiveness of the attacks and prove DoS attacks using ACK frames has more negative effects than CTS and RTS. They explain a non-cryptography solution does not help to solve these types of attacks. They also discuss the cost of encryption either symmetric or asymmetric is too high and is not worth in this case. Therefore they propose a hmac-sha1 based algorithm to authenticate control frames. Using NS2 simulation they prove the proposed model can enhance security of WLAN in case of such attacks.

In [17] the authors investigate two types of DoS attacks over mobile Ad hoc networks which are DoS attacks using control frames by one single attacker in compare to several attackers. They propose a packet by packet encryption method to counter the attacks. In their proposed model they also consider replay attack protection. They introduce a timestamp to confirm freshness of every received control frame. They explained if the length of timestamp is not long enough to grantee the freshness a sequence number can be added. Unlike timestamp the sequence number increments by one in each control frame transmission. Their method cannot provide security for broadcast packets where there is no control frames.

How attacker can leverage RTS to propagate the attack further by legal users is discussed in [12]. They propose a statistical approach to only detect not prevent such attacks using distribution pattern of the received packets. They set duration field of RTS packets to the maximum value (32767 us) to increase the impact of the attacks. They present some graphs to show under normal condition (no attack) the median value of transmission for all nodes is a constant value but under attack condition this median is not constant and are vary largely from the constant values of a normal network for all transmitting nodes which can be used for the attack detection. They discuss the pattern in their other paper [13] in more details.

DoS attacks in many different forms over wireless networks are evaluated in [4]. To implement control frame attacks, they make a real testbed and present the amount of damage these attacks can bring to the WLAN. They repeat their experiment using a variety of NIC card and two different access points. Using NS2 they simulate their proposed model to place a limit on duration value field of control frames. By the model, any packet with a larger duration value is simply truncated to the maximum value allowable. They explained the method can avoid cost of cryptographic algorithm and also prevent a long channel reservation by attacker.

There are some other papers [2, 3, 6, 9, 15] consider DoS attacks in wireless network by using other unencrypted wireless frames such as management frames. All of the above studies prove DoS attacks have a high negative impact on the wireless networks. The attacks cause serious damage on WLAN and can bring the network to a complete halt.

In order to simulate a model to prevent these attacks the first step to do is to ensure accuracy of the applied simulation tool through developing the attacks. When we firmly confirm that the tool is working properly, then we can guarantee results from simulation of the model to prevent the attacks are near to the real world as well.

Therefore in this research first we design and develop a new extension module to simulate wireless DoS attacks in OMNet^{++} and calculate the metrics. Then we make a real testbed to measure the same metrics as a reference to compare with results of OMNet^{++} to confirm accuracy of the extension module in OMNet^{++} under heavy loads of wireless DoS attacks.

3 Control Frames DoS Attacks

Control frames in wireless network assist to delivery of data to destination. Acknowledgment (ACK) control frames confirm data reception by the receiver. Request to send (RTS) and clear to send (CTS) are used to mitigate hidden node problem [7]. The process of RTS/CTS is done when a sender is about to transmit data. Sender first transmits RTS frame to inform other wireless nodes it has data to send. In response to RTS, the receiver transmits CTS to inform existing wireless nodes about this transmission. All these three control frames include a duration field which defines required time for data frame transmission to show how long wireless channel is reserved by the sender. This field is 16 bits long with the maximum value of 32767 us. Since control frames are not supported by any security mechanism, an attacker can send a flood of forgery control frames to a victim with maximum value for duration field. By doing this the attacker reserves the channel busy as long as it is desired to keep all authorized users away from the network. Even when RTS/CTS process is disabled in a wireless network, according to the 802.11 standard all nodes must response to a received RTS or CTS to avoid hidden node problem. This lead to make wireless network entirely vulnerable and susceptible to DoS attacks.

4 OMNet^{++} Review

OMNet^{++} is an open source and modular simulation framework. An OMNet^{++} model consists of hierarchically nested modules that communicate by message passing. The top level module is called network module which contains one or more sub-modules. Each sub-module can include one or more other sub-modules and there is no limitation for depth of these nested modules. The active modules are called simple modules so that each one has a corresponding C^{++} code programming. One or more simple modules can be joined together to produce a compound module which unlike simple module does not include a C^{++} code. Modules communicate via messages. These messages are sent either through gates or directly to the destination modules. Gate can be input or output for receiving or sending messages respectively.

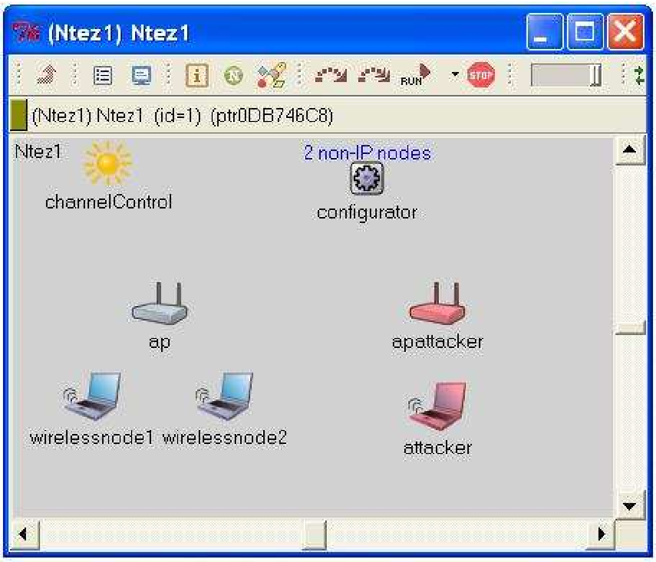

Figure 1: Structure of our simulation design

The simulation time advances whenever a module receives a message from other module or itself which is called self-message. Self-messages are used by any module to schedule events at a specific time. To define structure of a model, OMNet^{++} provides a topology description language which is termed as NED. A simulation model consists of at least one NED file includes network, simple, and compound module definitions [14].

5 Simulation Model Structure

In order to design our extension module and simulate control frames DoS attacks, we consider our simulation topology in two areas: legal area belongs to legal wireless network comprises of two wireless nodes (wireless-node1, wireless-node2) associated with an access point (ap), attacker area belongs to wireless attacker (attacker) associated to its own access point (ap-attacker). We include a channel control and a configurator to establish communication channel and assign ip address to the nodes respectively. A snapshot of running our simulation structure is presented in Figure 1.

In our simulation we consider wireless-node1 as sender and wireless-node2 as receiver. We simulate the traffic transmitted from sender to receiver in two patterns: connectionless traffic and connection oriented traffic. By doing this we can study if the type of traffic between nodes has any role in a network infected by DoS attacks. For connection oriented traffic we simulate TCP transmission and for connectionless traffic we simulate UDP.

ICMP is connectionless because it does not require nodes to handshake before establishing a connection. We simulate ICMP to obtain results of packet lost due to the attacks. We are also curious about if sender and receiver show the same reaction to the attacks, therefore we run our simulation so that the attacker can lunch each attack over sender, receiver, and access point separately to observe the results.

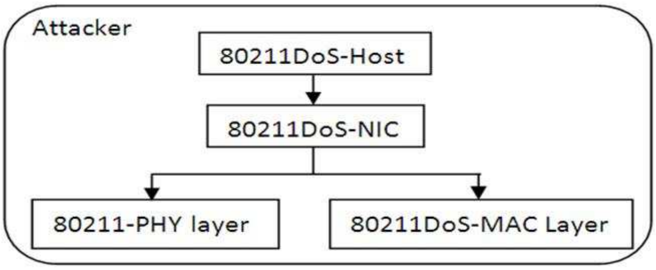

Figure 2: Structure of the new extension module

Table 1: System parameters

System parameters	Parameter Value
SIFS	10 us
DIFS	50 us
Channel bit rate	11 Mbs
Retry limit	7
CWmin	31
Slot time	20 us
Basic bit rate	2 Mbs
MAC header	224 bits
PHY header	192 bits

In order to design the new extension module we need to define a new network interface card (NIC). Therefore we create a new wireless host and we call it 80211 DoS-Host which has an 80211 DoS-NIC and then we add 80211 DoS-Host.ned and 80211 DoS-NIC.ned to OMNet^{++}. The new host includes a new MAC layer to develop DoS attacks. We have written the new MAC layer in C^{++} code and add it to the OMNet^{++} as a simple module which is called 80211 DoS-MAC. We let the PHY layer remains intact. The structure of this new attacker host is shown as Figure 2.

Our new MAC layer has ability to make forgery control frames to start DoS attacks in a basic bitrates according to IEEE 8021 standard [7]. In our configuration file our topology area size is set to 600 × 400 m^2 to cover all wireless nodes and access points. All the attacks through simulation start at 30th and stop at 60th second to have attack duration same as our real testbed. Inter arrival time and duration field of forgery frames are set to every 0.005 seconds and 32767 us respectively which are exactly the same as our real testbed to ensure enough fairness in the condition for comparison.

The system parameters and values for our simulation environment are reported in Table 1.

5.1 Performance Measures

In order to characterize influence of the attacks, in the simulation we investigate three metrics which are defined as follows:

- End-to-End delay is considered as average amount of time taken by a TCP or UDP packet to travel from originating node (wireless-node1) until it is successfully received at destination node (wireless-node2).
- Throughput is computed by dividing amount of data received by destination with the time taken to arrive at this node.
- We measure packet loss ratio as average number of packets discarded divided by the total number of packets during data transmission.

6 Simulation Results and Analysis

To obtain results of our experiments, in both testbed and simulation model, we focus our attention to five scenarios as follows:

- **Scenario A**: TCP throughput disruption due to control frames attacks;
- **Scenario B**: UDP throughput disruption due to control frames attacks;
- **Scenario C**: TCP end-to-end delay disruption due to control frames attacks;
- **Scenario D**: UDP end-to-end delay disruption due to control frames attacks;
- **Scenario E**: dropped ratio due to control frames attacks.

For each scenario we study effect of the attacks over three different victims as: access point, sender, and receiver to observe how different nodes with different functions show reaction to the attacks. This is important from the view point of attacker to determine the victim so that impact of the attacks reaches to the highest possible level.

6.1 Scenario A

In this scenario we transmit 1000-byte CBR TCP traffics from source to our sink using TCPSessionApp along TCPSinkApp modules. Then in our new MAC layer, we make our forgery control frame so that setReceiverAddress is set to the victim MAC address. We also assign an unknown MAC address to setTransmitterAddress to hide real MAC address of the attacker. We transmit all forgery control frames in interval times of 0.005 seconds with the maximum duration value. We schedule a new message called *dosstart* at 30th seconds which indicate start time of the attacks. The attacker station was configured to carry out the attacks periodically over the 30 to 60 seconds interval of the simulation. Figure 3 captures the results of the simulation run in this scenario.

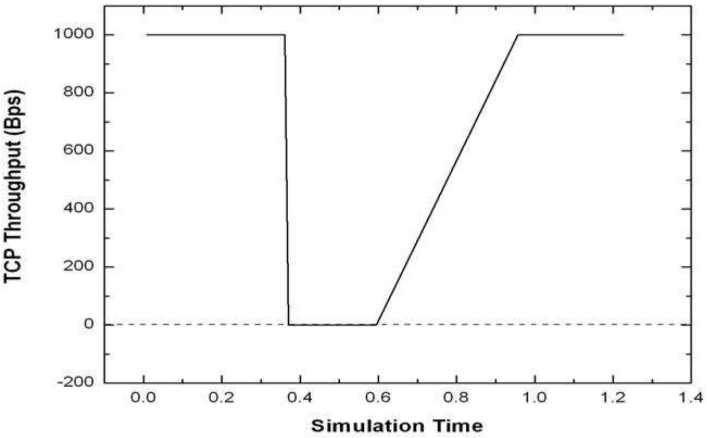

Figure 3: Impact of control frames DoS attacks over TCP throughput

We run the scenario several times while attacker targeting different victims as access point, sender, and receiver. The TCP throughput pattern was the same in all control frames attacks including ACK, CTS, and RTS. As the above graph shows, before attacker starts its activity the TCP traffics exchange in a normal way. When connection established constant data rate of 1000-byte TCP packets exchanged between nodes for 30 seconds.

Immediately after starting the attack at 30 seconds, traffic transmission between wireless nodes stopped for about 30 seconds attack duration. After finishing the attack at 60 seconds, while we expected to see normal transmission, we observed simulation time immediately increased to about 95 seconds so that there was no transmission during this gap time. After looking at the log file we conclude this gap time is related to duration filed of the last spoofed frame to reserve the channel. After this gap time when the media became free from the attacks, we could observe normal TCP traffic transmission between wireless nodes. We modify TCPSessionApp and TCPSinkApp modules to calculate amount of throughput during simulation time. Throughput before the attacks was about 226746.5418 Bps while during the attacks throughput reaches to null. After the attacks throughput back to normal and in average was about 233180.0971 Bps. From the graph and results we observed the attacks could easily bring the network down and create such loss of productivity.

6.2 Scenario B

Unlike last scenario, here we modify UDPBasicApp.ned file to send variable length UDP traffic to the sink using exponential distribution of 1000 bytes packet size. We also modify UDPBasicApp and UDPSink modules to collect required statistics. We collect the results during

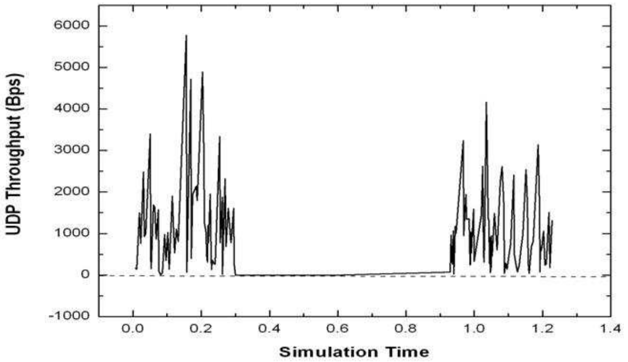

Figure 4: Impact of control frames DoS attacks over UDP throughput

simulation first for 30 seconds normal UDP transmission. Then at 30th seconds attacker station is scheduled to send its forgery control frames to floods access point, source, and sink separately for about 30 seconds. The results of these attacks are presented in Figure 4.

We computed throughput for 30 seconds normal transmission before the attacks which was about 251474.25 Bps. When the attacks start, immediately UDP transmission stopped and media was kept busy by the attacker for about 30 seconds. Our collected statistics show a null throughput during the attacks. When the attacks stopped, we observed like TCP simulation time increase very fast to about 92 seconds while unlike TCP traffic, during this gap time UDP packet transmission was done. The sender was able to transmit UDP traffic because of connectionless nature of UDP packets. However all these traffics entered the queue and were unable to reach to destination because the huge amount of forgery frames consume useful resources of the network which results in all legal communication failed. We saw as the gap time through simulation increased toward 92 seconds, the UDP traffics transmit to the destination as well. This leads to overflow the buffer size and queue became full therefore a large number of UDP packets dropped. After passing the duration of last spoofed frame, normal UDP transmission resumed in the network and average of throughput was about 246979.43 Bps.

6.3 Scenario C

In order to figure out effect of control frames DoS attacks on performance of wireless network in term of end-to-end delay, in our simulation model we consider to calculate amount of delay in both connection-oriented and connectionless in either constant and variable data rate. Therefore in this scenario we use TCPSessionApp along TCPSinkApp module to make session between source and sink. We modify existing OMNet^{++} MAC layer of wireless-node1 and wireless-node2 to compute delay in MAC layer. Attacker starts its attacks at 30 seconds after simulation starts. The increase in amount of delay because of the attacks is shown in Figure 5 as follow.

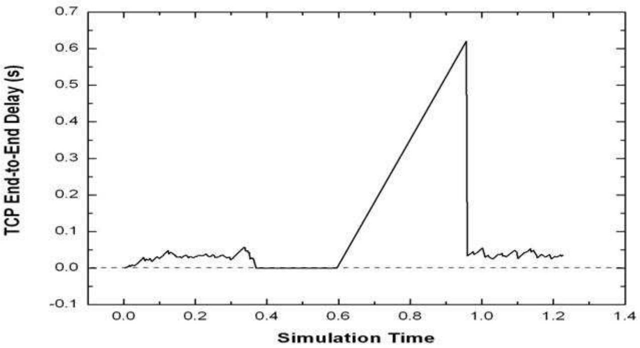

Figure 5: Impact of control frames DoS attacks over TCP end-to-end delay

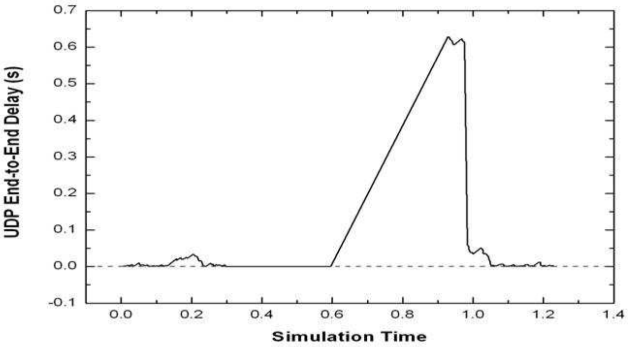

Figure 6: Impact of control frames DoS attacks over UDP end-to-end delay

Before the attacks, average delay was small about 0.0279 seconds. During the attacks we did not observe any traffic transmission therefore there was no any delay as well. Since during the attack TCP connection was disconnected, therefore after the attacks source and sink try to make connection from the beginning. We observed a large number of already scheduled frames transmitted in a short time which make them enter the queue. Therefore for a short period of time we had a peak in amount of delay. When all existing frames in the queue proceeded, the amount of delay backed to normal. The average amount of delay after attack was about 0.0455 s.

6.4 Scenario D

In this scenario we modify MAC layer of source and sink so that we can calculate end-to-end delay for variable data rate UDP packets transmission during simulation time. We observed almost the same pattern for delay of UDP packets with variable length and delay of TCP packets with constant length. The impact of the attacks on delay of UDP packets is shown in Figure 6.

Before the network goes under the attack, we computed the average end-to-end delay in MAC layer between wireless-node1 and wireless-node2 which was small about

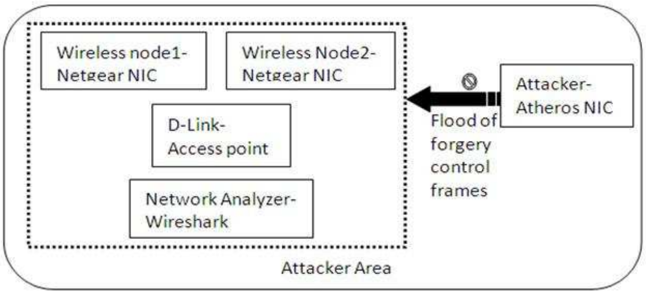

Figure 7: Testbed setup environment

0.0091 seconds. During the attacks there was no any UDP traffic exchange therefore there was no any delay. As it mentioned before, during the gap time, a large number of UDP packets transmitted so that some of them entered the queue and some of them dropped. The packets entered the queue remained there for a long time until the attacks finished therefore delay for these packets increased so much. After the attacks finished we observed the peak in delay time because the network started to process these queued packets. The computed delay after attack was about 0.1288 seconds.

6.5 Scenario E

In this scenario we attempts to obtain results of packet lost ratio when the wireless network is under control frames flooding attacks. In the simulation model we use pingApp module over wireless-node1 as sender of ping-requests and wireless-node2 which answers to the requests with ping-replies. We set packet size to 56B and one seconds intervals as it is in real network. The result shows from 124 sent packets, about 47 packets dropped during the attacks which shows about 37.9032% packet lost ratio during the attacks.

At the same time we calculated the round trip time. This metric shows the time required for a packet to travel from a specific source to a specific destination and back again. It was about 0.0023 seconds before the attack and then it reached to zero during the attack time like delay. After the attacks for a short time, the round trip time of ICMP packets shows a sharp increase to transmit enqueued packets then back to normal. The average amount of round trip time after attack was about 0.2224 seconds.

7 Testbed Design

We obtained our simulation results from OMNet^{++}. Now in order to validate accuracy of our extension module we compare these results with a real wireless network. Therefore we make a simple testbed with structure presented in Figure 7.

In our previous work [10], we made a real testbed to investigate control frames DoS attacks using Linksys wireless access point. Results proved the attacks over clients were completely successful but the Linksys was not susceptible to the attacks. Therefore we decided to repeat the experiments with another access point to investigate the network behavior in case of such attacks. This time we chose a D-Link wireless access point and two wireless clients connected to the D-Link.

We design our testbed to be exactly like our simulation design. The rationale behind this choice is that since we want to investigate accuracy of OMNet^{++} by using the extension module in compare to real world, therefore similar conditions in both environments can provide more reliable and fair results which make comparison more accurate. Therefore we design the testbed with two wireless clients both with Netgear chipset connected to wireless D-Link access point. Attacker includes an Atheros chipset which is configured in monitor mode to be able to start the attacks. We use Wireshark to capture all exchanged traffics over all the channels to investigate network behavior during the attacks.

We make our forgery ACK, RTS, and CTS frames using *Khexedit* in *Ubuntu* and transmit them to the victim in an attack rate exactly like we did in our simulation. The employed application to flood targets is *file2air* tool.

8 Experimental Results

In order to measure our performance metrics we consider three scenarios in our experiments as follows:

- **Scenario A**: ACK DoS attack over wireless clients and access point (ACK-C, ACK_AP);
- **Scenario B**: CTS DoS attacks over wireless clients and access point (CTS-C, CTS-AP);
- **Scenario C**: RTS DoS attack over wireless clients and access point (RTS-C, RTS-AP).

In all experiments, the experiment time is divided in three parts: before, during, and after the attack. The process of experiments have been designed so that first we monitor the wireless network under normal conditions with no attacks for about 30 seconds and after that we start the attacks. After the attack period, we let the wireless network back to normal transmission for about 30 seconds. During these times we monitor the network performance and collect our metrics to compare with our simulation results previously specified.

8.1 Scenario A

In this experiment we made our forgery ACK frame with maximum value of duration filed and transmit it continually to the D-Link access point and wireless nodes separately. We start the attacks at 25 seconds and stop it at 56 seconds to have the same attack duration like our simulation. The results of these attacks over access point and clients are presented in Figure 8 and Figure 9 respectively.

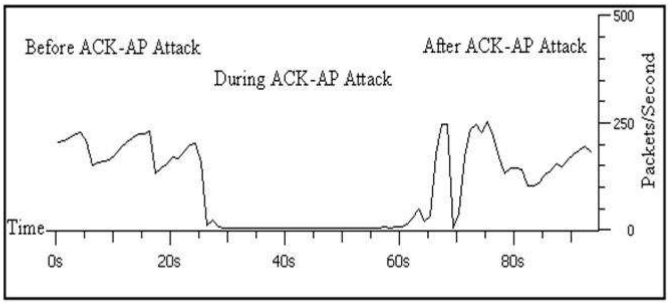

Figure 8: Impact of ACK DoS attack over access point

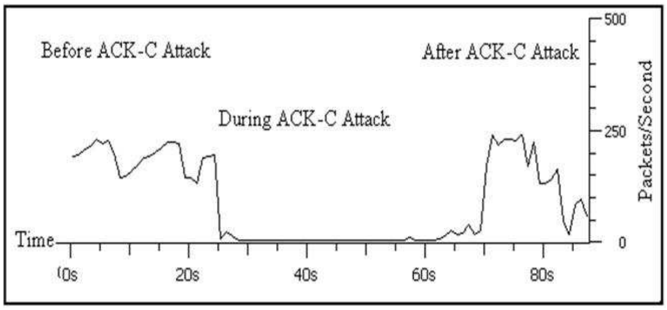

Figure 9: Impact of ACK DoS attack over wireless clients

As the above figures show, neither target access point nor target clients resist the attack and completely blocked during the attack time. The wireless clients were not able to access to the network and our D-Link access point unlike Linksys in [10] was completely overwhelmed by the attacker forgery frames. To compare with OMNet^{++} results, we measure throughput and packet lost ratio. The results are shown in Table 2.

Comparing throughput from the above table with both TCP and UDP throughput simulation results prove the simulation results are quite close to the real world measurements. The results clearly indicate a good performance of the simulation and the extension module. Also by comparing packet lost ratio from the above table with simulation results, we can see the packet lost ratio in OMNet^{++} is almost the same as real testbed. We repeated our experiments several times as we did in our simulation and the results were identical.

Table 2: Experimental results of ACK DoS attacks

Attack	Throughput (Bps)			Lost
	Before	During	After	Ratio (%)
ACK-AP	204972.6	0	132315.5	40
ACK-Client	203165.6	0	93675.9	39

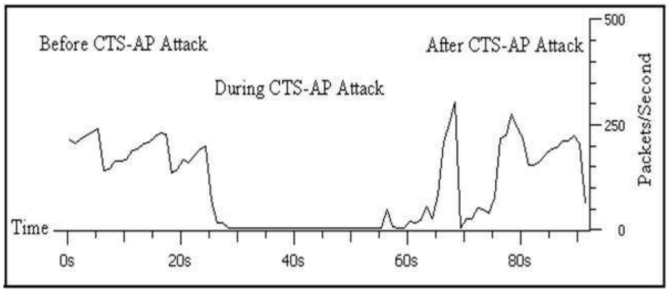

Figure 10: Impact of CTS DoS attack over access point

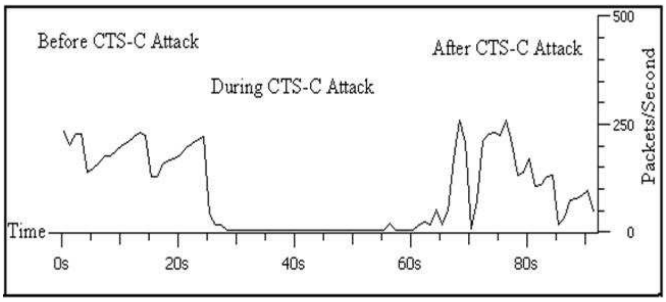

Figure 11: Impact of CTS DoS attack over wireless clients

8.2 Scenario B

To run this experiment we made a forgery CTS control frame and set the maximum duration value for it. Then we continually injected the forgery frame to the target access point and clients. The result of these attacks over access point and clients are shown in Figure 10 and Figure 11 respectively.

As the above graphs show the attack was completely successful over both access point and client. The figures show our D-Link access point unlike Linksys in [10] was totally busy to manage forgery frames and could not respond to any other legal transmission. During the attack all useful buffer of D-Link consumed by useless forgery packets which made the access point to disconnect other legal clients and fail to provide any services for its authorized stations. Results of monitoring the network to measure our metrics are presented in Table 3.

From the above table we note that the simulation results match our measurement results perfectly. This indicates a correct implementation of the simulation environment using the extension module. Both results confirm

Table 3: Experimental results of CTS DoS attacks

Attack	Throughput (Bps)			Lost
	Before	During	After	Ratio (%)
CTS-AP	205714.1	0	126620.2	42
CTS-Client	203656.1	0	78807.8	41

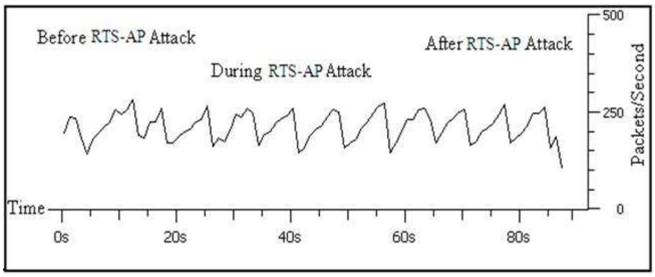

Figure 12: Impact of RTS DoS attack over access point

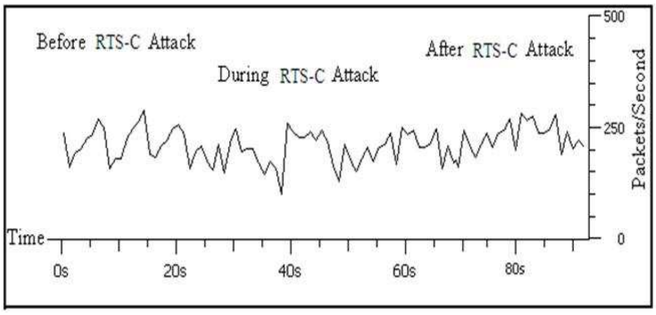

Figure 13: Impact of RTS DoS attack over wireless clients

effectiveness of the attacks to degrade throughput of wireless network as we can see a null throughput and high lost ratio during the attacks.

8.3 Scenario C

In this scenario, we created congestion by continually injecting our forgery RTS frames over target clients and access point. The results of this attack are shown in Figure 12 and Figure 13 respectively.

As we can see from the above figures this time the attack was not successful at all. Like Linksys access point in [10] D-Link was resistance to RTS attack. The access point easily manages to handle all our forgery RTS frames besides of all other legal frames. We also observe the attack was not successful over wireless clients as well. To confirm the results we repeated our experiment several times with different attack rate. We even increase the experiment time to see if network can be finally down during a longer attack period. But the results were the same and the attack was not successful at all. We collected the performance metrics to observe network behavior statistically. The results are shown in Table 4.

As we can see from the above table, throughput was not degrade by the attack. Also drop rate ratio show null during our experiments. Comparing above results to our simulation results shows a complete mismatch. The simulation results by using extension module prove the RTS attack like other attacks was successful over both client and access point. However this cannot entirely deny accuracy of the module and simulation environment because in real world some wireless devices do not properly implement the 802.11 MAC specifications and improperly reset their duration values [4].

Table 4: Experimental results of RTS DoS attacks

Attack	Throughput (Bps)			Lost
	Before	During	After	Ratio (%)
RTS-AP	208243.2	0	206620.6	0
RTS-Client	206782.2	0	498669.3	0

Besides comparing experimental results that were obtained from our previous work [10] with our above experimental results shows even results in different real testbeds can show a significant difference. We believe that the reason behind this is different implementation of IEEE 802.11 standard is used by various manufacturers [5].

9 Conclusion

DoS attacks using unprotected control frames can severely degrade wireless network performance. Therefore providing security mechanism to prevent these attacks is a requirement for wireless networks. In this work we design and develop a new extension module for OMNeT^{++} tool to implement wireless DoS attacks. We evaluate the reliability and accuracy of this module in compare to a real testbed. We were able to make a wireless testbed identical to our simulation model to have fair enough conditions to compare the experimental and simulation results.

Both testbed and simulation results prove that DoS attacks by exploiting unprotected control frames can highly degrade wireless network performance. Based on the experiments and corresponding results we determine that OMNeT^{++} using the extension module performs quite well. This can guarantee that simulation of models using this simulation tool to prevent the attacks can provide results near to real world. As our future work we are developing a per-frame authentication model which provides three improvements over the current control frames as: authentication, integrity and freshness check. In the model, by using envelop hash function and a secret key along with a 32-bit timestamp, the recipient can verify validity of the received control frames and determine the forgery frames to discard.

References

[1] A. Agah, K. D. Sajal, “Preventing DoS attacks in wireless sensor networks: A repeated game theory approach,” *International Journal of Network Security (IJNS)*, vol. 5, no. 2, pp. 145-153, 2007.

[2] B. Aslam, M. Akhlaq, S. A. Khan, “IEEE 802.11 wireless network simulator using verilog,” *Proceedings of the 11th WSEAS International Conference on Communications*, pp. 393-398, Greece, 2007.

[3] B. Aslam, M. H. Islam, S.A. Khan, "Pseudo randomized sequence number based solution to 802.11 disassociation denial of service attack,"*Proceedings of the 1st International Conference on Mobile Computing and Wireless Communications (MCWC 2006)*, pp. 215-220, Amman, Jordan, 2006.

[4] J. Bellardo, and S. Savage, "802.11 denial of service attacks: Real vulnerabilities and practical solutions," *Proceedings of the 12th USENIX Security Symposium*, pp. 15-28, Washington DC., USA, 2003.

[5] M. Boulmalf, E. Barka, and A. Lakas, "Analysis of the effect of security on data and voice traffic in WLAN,"*ACM journal of Computer Communications*, vol. 30, no. 11-12, pp. 2468-2477, 2007.

[6] B. Chen, and V. Muthukkumarasamy,"Denial of Service Attacks against 802.11 DCF," *Proceedings of the IADIS International Conference, Applied Computing*, pp. 5, San Sebastian, Spain, 2006.

[7] IEEE Computer Society, *Wireless LAN Medium Access Control and Physical layer Specification*, 1999.

[8] P. Ding, "Central manager: A solution to avoid denial of service attacks for wireless LANs," *International Journal of Network Security*, vol. 4, no. 1, pp. 35-44, 2007.

[9] Y. S. Lee, H. T. Chien, and W. N. Tsai, "using random bit authentication to defend IEEE 802.11 DoS attacks," *Proceedings of the 5th WSEAS International Conference on Data Networks, Communications and Computers*, pp. 1458-1500, Bucharest, Romania, 2006.

[10] M. Malekzadeh, A. Azim A. G., D. Jalil, and S. Shamala, "Empirical analysis of virtual carrier sense flooding attacks over wireless local area network,"*Journal of Computer Science (JCS)*, vol. 5, no. 3, pp. 214-220, 2009.

[11] A. Rachedi and A. Benslimane, "Impacts and solutions of control packets vulnerabilities with IEEE802.11 MAC," *Wireless communications and mobile computing*, vol. 9, no. 4, pp. 469-488, 2009.

[12] K. Sugantha and S. Shanmugavel, "Anomaly Detection of the NAV attack in MAC layer under non-time and time-constrained environment,"*IEEE International Conference on Wireless and Optical Communications Networks (IFIP '06)*,pp. 1-5, Bangalor, 2006.

[13] K. Sugantha, and S. Shanmugavel, "A statistical approach to detect NAV attack at MAC layer," *Proceedings of the International Workshop on Wireless Ad-Hoc Networks*, pp. 6, London, UK, 2005.

[14] OMNET++Discrete Event Simulation System. (http://www.omnetpp.org/doc/omnetpp40/manual/usman.html)

[15] Z. I. Qureshi, B. Aslam, A. Mohsin, and Y. Javed, "Using randomized association ID to detect and prevent spoofed ps-poll based denial of service attacks in IEEE 802.11 WLANs," *ACM WSEAS Transactions on Communications*, vol. 7, no. 3, pp. 170-179, 2008.

[16] W. Ren, "Pulsing RoQ DDoS attacking and defense scheme in mobile ad hoc networks," *International Journal of Network Security*, vol. 4, no. 2, pp. 227-234, 2007.

[17] Y. Zhou, D. Wu, and S. M. Nettles, "Analyzing and preventing mac-layer denial of service attacks for stock 802.11 systems," *Proceedings of IEEE/ACM First International Workshop on Broadband Wireless Services and Applications (BWSA'4)*, pp. 1103-1106, CA, USA, 2004.

M. Malekzadeh Received the B.Sc in Mathematics Computer Science in 2000 from S.B University in Iran. During 2000-2005 she was the head of quality control in NICICO which is one of the biggest mine copper in the Middle East. She received her M.Sc in software engineering from University Putra Malaysia (UPM) in 2007. She is now a Ph.D. student in computer security. Her main research interests include computer networking, wireless communication, and security in computer.

A. A. Abd Ghani Received the B.Sc in Mathematics Computer Science from Indiana State University in 1984 and M.Sc in Computer Science from University of Miami in 1985. He received the Ph.D. in Software Engineering from University of Strathclyde in 1993. He is an Associate Professor and the Dean of Faculty of Computer Science and Information Technology, University Putra Malaysia. His research interests are software engineering, software measurement, software quality, and security in computing.

S. K. Subramaniam Received the B.S. degree in Computer Science from University Putra Malaysia (UPM), in 1996, M.S. (UPM), in 1999, PhD. (UPM) in 2002. Her research interests are Computer Networks, Simulation and Modeling, Scheduling and Real Time System.

J. M. Desa is head of Network Transmission and Security, and also an Associate Principal Researcher at Telekom Research and Development Malaysia. Dr. Jalil received his Bachelor of Electronic and Electrical Engineering (Hons) from University of Portsmouth (UK), and his PhD from University of Strathclyde (UK). His major research fields are computer network, Internet protocols (IPv4/IPv6), routing, security, network management system, and router design.