Jamming Vulnerabilities of IEEE 802.11e

David J. Thuente, Benjamin Newlin, and Mithun Acharya

***Abstract*— It has long been recognized that complete jamming of wireless networks can be realized by generating continuous noise with sufficient power in the vicinity of the wireless network. Recent work has shown that for IEEE 802.11b similar jamming effectiveness can be achieved using "intelligent" or protocol specific techniques. These techniques support jamming with low energy requirements and low probability of detection. This paper extends those results to IEEE 802.11e. Many attacks that worked on 802.11b can be applied to 802.11e. Some of them are not nearly as effective as they were for 802.11b, while others are more effective. Other jamming techniques specific for IEEE 802.11e are also discussed. We use OPNET to first study the effects of periodic jamming on the network throughput. The critical step for jamming effectiveness is adding *intelligence* to the jammer by using knowledge of the protocol and exploiting crucial intervals, control messages, and mechanisms. We discover and analyze new vulnerabilities using intelligent jamming in 802.11e. By exploiting the priority levels, we show that "misbehaving" nodes can be used to jam 802.11e by decreasing either the access time or contention window. We also show that the priority level is crucial in this attack. We document the areas of increased vulnerability of this protocol relative to 802.11b. Most importantly, we show that 802.11e can be "jammed" by nodes that obey all protocol rules and use only normal traffic.**

***Index Terms*— Denial of Service (DoS), IEEE 802.11e, MAC protocol attacks, intelligent wireless jamming, legal attacks.**

I. Introduction

Wireless networking is becoming increasingly more prominent in communication technology. The flexibility and mobility afforded by wireless communication make it a superior option to wired networks in areas ranging from computers and PDAs to cell phones and accessories. Wireless local area networks are used extensively in offices, campuses, and businesses. However, wireless transmissions cannot be directed without losing the very mobility that makes them useful. Radio transmissions in wireless networks are broadcast instead, meaning that any receiver within range will be able to hear the transmission. It also means that any other active transmitters in the same frequencies in the area will cause interference with the signal, which can cause errors in the data or corrupt it entirely. These simultaneous transmissions are called *collisions* and usually cause the data from both sources to be lost.

In order to reduce the number and frequency of collisions in areas where several stations have data to transmit, it is necessary to formulate a Media Access Control (MAC) protocol to be used by all stations that will divide access to the wireless medium between stations. The IEEE 802.11 family of specifications does this by using Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) scheme. There have been several amendments to the original standard to add functionality for higher data rates and quality of service. This paper focuses on amendments b and e.

While the MAC in 802.11 works very well under ideal conditions, in cases where there is emergency or military data being transferred, there might always be malicious users who wish to stop or severely hamper communication. This act is called *jamming* and the station or transmitter performing it is called a *jammer*. Because simultaneous transmissions collide, jamming can be easily accomplished by constantly transmitting a high power signal. However, such a transmitter could be easily detected and traced to its source. This technique always requires large amounts of power.

It would be preferable to minimize the jammer's signal and transmit them strategically in such a way that they do not completely block communication, but instead attack the access control mechanism. This technique would time the jammer's transmission to target key messages so that communication is not allowed by the MAC even though the wireless medium may be available. Jamming stations that utilize knowledge of the MAC protocol to block communications are called *intelligent jammers* and are the focus of this paper.

The rest of the paper is organized as follows: First, there will be a brief overview of access control in 802.11b and the differences and additional functionality in 802.11e. Then the simulation model used for investigating the jamming methods will be presented. Previous jamming techniques and results in 802.11b will be summarized and these are followed by their counterpart techniques in 802.11e. Finally, jamming vulnerabilities specific to 802.11e will be discussed and analyzed with emphasis on the priority levels. Finally, effective jamming techniques that conform entirely to the 802.11e protocol will be documented.

David J. Thuente is a Professor of Computer Science, North Carolina State University, Raleigh, NC 27695 USA, phone: 919-515-7003, fax: 919-513-1895, Email: thuente@csc.ncsu.edu.

Benjamin Newlin is currently a Software Engineer at Nortel, Inc., 4004 E Chapel Hill-Nelson Hwy, Research Triangle Park, NC, 27709. Email:bcnewlin@nortel.com. This work was part of an independent research project at North Carolina State University.

Mithun Acharya is currently a research intern at Microsoft Center for Software Excellence, Redmond, and is an advanced PhD student with the Department of Computer Science, North Carolina State University, Raleigh, NC 27695 USA, Email: acharya@csc.ncsu.edu.

1-4244-1513-06/07/$25.00 ©2007 IEEE

II. MAC Layer Functionality

A. IEEE 802.11b

This paper focuses on the MAC layer controls for the 802.11 family of specifications for MAC and physical layer (PHY) communications between wireless stations. The basic access mode in 802.11b relies on an implementation of the CSMA/CA scheme known as the Distributed Coordination Function (DCF). The DCF uses timing techniques involving Inter-Frame Spaces (IFS) and exponential backoff to avoid collisions and ensure equal throughput to all stations.

Any station that wishes to transmit must first listen, or sense, the wireless medium to determine if it is busy. If the medium remains free for the period of a DCF Inter-Frame Space (DIFS) then the station may transmit immediately. Otherwise, if the medium is busy or becomes busy during the DIFS, the station enters a contention period in which it will utilize the exponential backoff mechanism. It will randomly select a backoff value from within the Contention Window (CW) and continue monitoring the state of the medium. As soon as the medium remains free for a DIFS, the station will begin to decrement its backoff timer once for each time slot the medium is free. A time slot is a fixed length of time specified by the protocol. If the backoff timer reaches zero, the station will transmit. If the medium becomes busy during the backoff period, the station will freeze the timer at its current value and resume countdown when the medium has been free for a DIFS again. Each station is also required to perform backoff after a successful transmission to avoid one station monopolizing the medium.

Each transmission requires an acknowledgment (ACK) from the receiving station upon arrival. If the sending station does not receive the ACK after transmitting, it assumes that a collision with another transmission has occurred. In this instance, the sender will double the size of its CW and start a new backoff by selecting a number from the new, larger, contention window. This is referred to as binary exponential backoff and it helps the network recover from frequent collisions by further spacing out future transmissions. The basic DCF access method is well known and is covered in [8]. IEEE 802.11b also employs an optional Request-To-Send/Clear-To-Send (RTS/CTS) mechanism. Its functionality is outlined in Fig. 1.

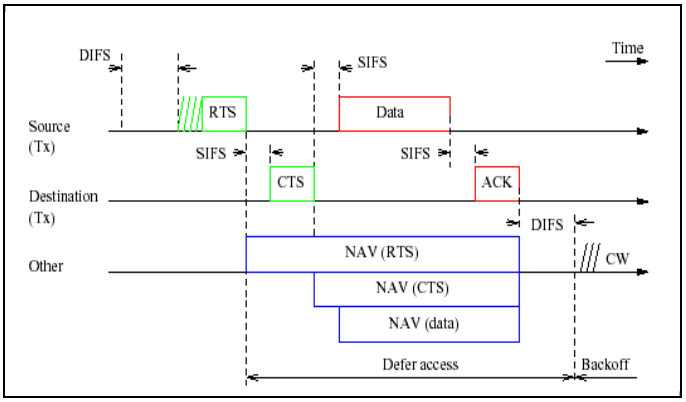

Fig. 1: Access using RTS/CTS method.

B. IEEE 802.11e

802.11e is based on the original access mechanisms for 802.11b but has added functionality for Quality of Service (QoS) guarantees. Data transmissions that contain voice or video data can be very time-sensitive. QoS is aimed at ensuring a lower delay to access the medium for this high priority traffic. This is incorporated on top of the 802.11b DCF in order to maintain backward compatibility.

Priority of data is specified by an Access Category (AC). Each data packet sent must be assigned an AC. The standard defines four AC's, in order from highest to lowest priority: Voice, Video, Best Effort, and Background. Best effort is the standard priority and all data from a station using 802.11b will be assigned this AC at an 802.11e station.

The protocol which controls medium access in 802.11e is the Enhanced Distributed Coordinated Access (EDCA). The EDCA operates in much the same way as the DCF with a few exceptions. The first method for implementing QoS is the addition of a new parameter called the Arbitration Inter-Frame Space (AIFS). This replaces the DIFS used in 802.11b as the length of time the medium must be free before a station can transmit or resume backoff. The AIFS is determined by its AIFS Number (AIFSN) as specified in the standard for each AC. In this way, higher priority traffic is allowed to access the medium or resume backoff sooner than lower priority. The standard AIFSN for the voice and video traffic is 2, which makes their AIFS equal to the standard DIFS in 802.11b. The relationship between several IFS times is illustrated in Fig. 2.

The second method of implementing QoS functionality is the Contention Windows (CW). As in 802.11b, the backoff value is randomly selected from the current CW. In the event of a collision, the CW is doubled - up to a specified maximum value. In 802.11e, the minimum and maximum values for the CW are lower for high priority AC's. These AC's spend less time performing backoff after each collision or successful transmission. Higher priority AC's are able to transmit more frequently, on average, than lower priorities since a backoff is always performed after a successful transmission.

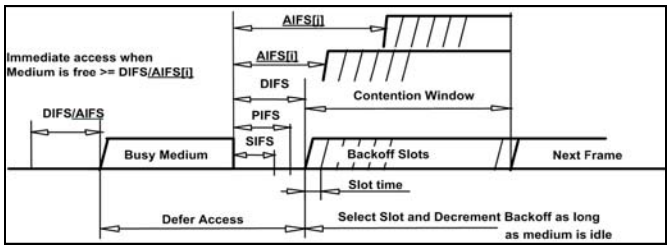

Fig. 2: Relationships of key IFS times.

The AC determines more than just the values for these parameters. In fact, each station maintains a separate queue for each AC, which act independently of each other. Each maintains its own backoff timer and CW size. In the event that two AC queues within a station are ready transmit, the lower AC will always yield to the higher. This is known as an internal collision and it does not cause the CW size to be increased. There are several other optional functionalities in 802.11e including transmit opportunity and block ACKs that are not relevant to this paper.

III. Network Model

This section describes the basic model for the network used in analyzing the 802.11e protocol. The specific settings outlined below are for a normally operating network and many have been changed for analysis of different scenarios. Any changes to the basic settings will be noted during the analysis.

The network model used is identical to the model in [8]. It consists of nine stationary wireless nodes arranged in a circular pattern. In the center is an access point (AP) and a jammer node. This is depicted in Fig. 3. All nodes are well within the maximum transmission range which is specified in the standard as 300 meters. In fact, no node is more than 50 meters from any other, so there is no hidden terminal problem.

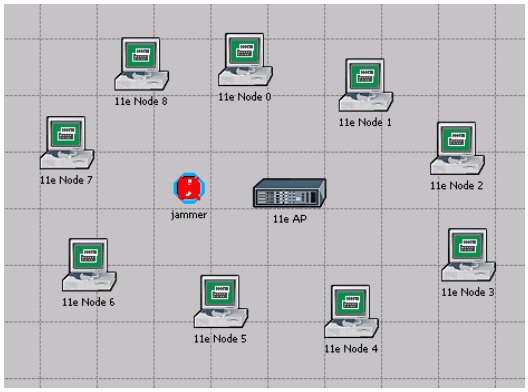

Fig. 3: Network topology.

All nodes and the AP are operating at the PHY specifications of 802.11e. They are utilizing Direct Sequence Spread Spectrum (DSSS) transmission with a data rate of 11 Mbps. Even though there is no hidden terminal problem, some of the techniques used in [8] require RTS/CTS. In order to be able to compare these results with those in [8], the RTS threshold is set to 128 bytes, enabling RTS/CTS for almost all data packets. Exponential traffic, mean inter-arrival time of .03 sec., is generated at each node. The size of each packet also follows an exponential distribution with a mean of 2000 bytes. These settings are summarized in Fig. 4.

Packets over the maximum size of 2304 bytes are discarded, so transmission will be attempted for approximately 68% of the packets generated, due to the exponential distribution. As shown in [8], the average offered load from each node is:

(2000 + 28 byte header) * 100/3 pkts/s * .68 * 8 bits/byte = 367.7 Kbit/s.

Then the total offered load for the network with nine nodes is:

367.7 Kbit/s * 9 nodes = 3.31 Mbit/s.

Due to the RTS/CTS mechanism and the fact that all packets must be relayed through the AP, the actual achievable throughput in this model is shown via simulation to be approximately 1.5 Mbps; see Fig. 6. The parameters for the EDCA used in analysis were set to the default values specified by the 802.11e standard; see Table I. The traffic from each node is distributed across the AC's in a set distribution algorithm that yields 35% Voice traffic, 35% Video, 20% Best Effort, and 10% Background. The rationale for this decision was based on the assumption that a network would not need to

Attribute	Value
name	11e Node 0
model	wlan_station_adv_dist
Destination Address	Random
Traffic Generation Parameters	(...)
Start Time (seconds)	constant (5)
ON State Time (seconds)	constant (1200)
OFF State Time (seconds)	constant (0)
Packet Generation Arguments	(...)
Interarrival Time (seconds)	exponential (.03)
Packet Size (bytes)	exponential (2000)
Segmentation Size (bytes)	No Segmentation
Stop Time (seconds)	Never
Traffic Type of Service	Best Effort (0)
Wireless LAN	

Fig. 4: Traffic model for wireless stations.

Attribute	Value
name	11e Node 0
model	wlan_station_adv_dist
Destination Address	Random
Traffic Generation Parameters	(...)
Traffic Type of Service	Best Effort (0)
Wireless LAN	
Wireless LAN MAC Address	Auto Assigned
Wireless LAN Parameters	(...)
BSS Identifier	Auto Assigned
Access Point Functionality	Disabled
Physical Characteristics	Direct Sequence
Data Rate (bps)	11 Mbps
Channel Settings	Auto Assigned
Transmit Power (W)	0.005
Packet Reception-Power Threshol...	-95
Rts Threshold (bytes)	128
Fragmentation Threshold (bytes)	None
CTS-to-self Option	Enabled
Short Retry Limit	7
Long Retry Limit	4
AP Beacon Interval (secs)	0.02
Max Receive Lifetime (secs)	0.5
Buffer Size (bits)	1024000
Roaming Capability	Disabled
Large Packet Processing	Drop
PCF Parameters	Disabled
HCF Parameters	Default

Fig. 5: Wireless attributes of a station node.

implement 802.11e QoS unless it had a large amount of high priority traffic. It is shown later that the results are largely independent of AC traffic loads. Fig. 6. shows the total throughput of the network and Fig. 7 shows network throughput per AC. Note that each AC comprises on average the same percentage of total throughput for traffic generation.

It is important to note that the QoS enhancements are not designed to ensure that a higher priority packet will always be transmitted before a lower priority one. Instead, the amendment

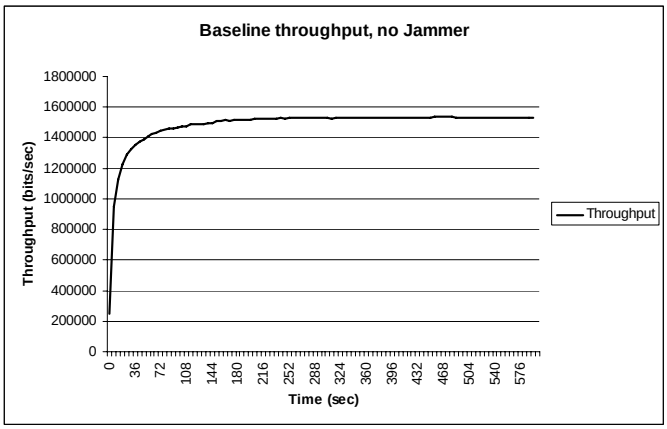

Fig.6: Baseline network throughput (bps).

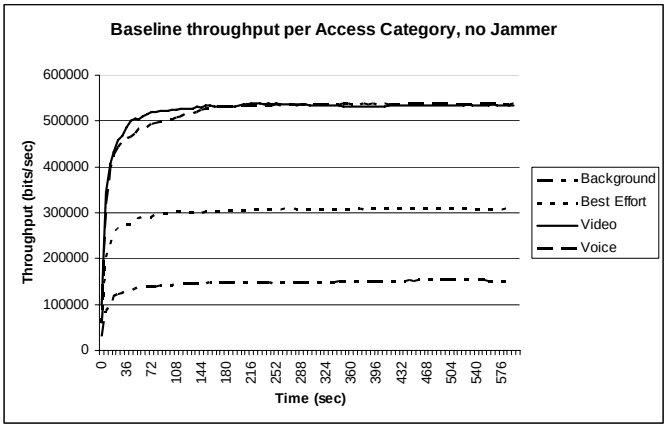

Fig.7: Baseline network throughput by AC.

Table I: Default parameters for EDCA in 802.11e.

	CW_{min}	CW_{max}	AIFSN
Voice	7	15	2
Video	15	31	2
Best Effort	31	1023	3
Background	31	1023	7

is focused on reducing the average media access delay for the higher priorities. With the distributions used here, the lowered delay for voice and video results in a higher throughput due to more frequent control of the medium.

IV. Jamming Techniques Used In 802.11b

In [8], [2], and [3] the authors covered several jamming techniques and their effectiveness in 802.11b in terms of network throughput and power expended by the jammer. The jamming techniques covered were divided into four categories: trivial jamming, simple periodic jamming, intelligent jamming, and misbehaving nodes. The results there showed that intelligent jamming was much more effective in decreasing throughput with minimal probability of detections or energy use. In the case of misbehaving nodes, the authors showed that manipulating access control and contention window parameters at one or two nodes could have serious effects on the entire network. The critical part of results there was the necessity to violate the access control and/or the contentions specifications of 802.11b.

For this paper, many of the previous techniques were applied to 802.11e to analyze if the MAC layer changes implemented mitigated the jamming effectiveness. Trivial jamming is simply producing continuous noise to prevent communication [8]. It was used as a baseline for comparison.

Simple periodic jamming uses timed pulses of noise to disrupt communication. The frequency of the pulses can be either a fixed value or can vary following some stochastic distribution algorithm. In [8], both fixed and exponentially distributed periods of jamming were tested. These techniques were found effective in relation to the frequency of the pulses: higher frequencies resulted in greater jamming effectiveness at the cost of power consumption. These tests were duplicated for 802.11e in order to establish the effect of the staggered AIFS times for each AC. It was not expected that the technique would be more effective, but rather that the larger delays to access the medium for some AC's could cause longer and more frequent periods of silence. These, in turn, would decrease the effectiveness of periodic jamming by increasing the likelihood of a pulse being sent while the medium was free. The results for the simple periodic cases are summarized in Fig.8. Throughputs are significantly decreased with periods as high as 2000 microseconds. While the results are similar to those obtained for 802.11b, the jamming was slightly less effective as had been anticipated because of the multiple ACs. The jamming pulse is one microsecond in both scenarios.

The intelligent jamming techniques given in [8] such as CTS and ACK jamming and RTS faking are essentially duplicates for 802.11e (except for AC implications) and are not reported here due to space limitations.

The major area of interest in this paper is the technique of misbehaving nodes, see [6]. In [8], a misbehaving node was simply one that did not perform a backoff or used a smaller window for its backoff. In the primary scenario there, the node would always attempt to transmit immediately after the DIFS period. While one misbehaving node was moderately successful at DoS, two could drop the network throughput to nearly zero. Due to the added parameters adopted in 802.11e for establishing QoS, there are many more ways for a node to "misbehave" by manipulating AIFSN and CW parameters outside of the specified ranges. For our purposes, the aim of these nodes is to jam the network and decrease throughput.

Misbehaving nodes are a form of greedy behavior and there are algorithms for access points that purport to detect various forms of this behavior. The techniques that reduce the access time can be detected by the access point or nodes that reduce the CW can be detected by an algorithm [5]. Neither technique detects the legal jamming given in the next section.

The majority of the techniques used in this research involved one or two nodes operating under EDCA parameters that had been manipulated in order to lower network throughput. The four parameters that were altered were:

- Priority distribution of packets generated
- AIFSN
- Minimum CW size
- Maximum CW size.

The misbehaving nodes are configured to generate smaller

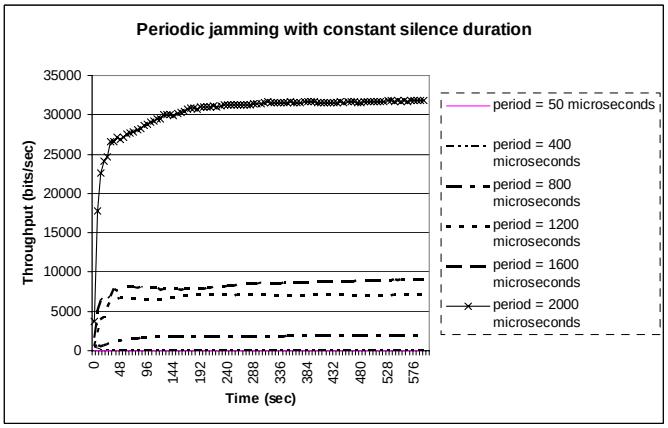

Fig.8: Throughput with simple periodic jamming.

packets at a higher frequency than the normal nodes. The average (exponential) packet size is 150 bytes and the average inter-arrival time is .005 seconds. The overall load offered by these nodes is slightly less than that of the normal nodes:

(150 + 28) * 100/.005 pkts/sec * 8 bits/byte = 284.8 Kbit/s.

The traffic generated by the other eight nodes is 4.33Mbps but packets over 2304 bytes are dropped so the effective offered load is 2.94 Mbps. These parameters are optimal for jamming because the point is to disrupt with short bursts, not continuously generate noise; that uses a lot of power.

The first techniques for misbehaving nodes analyzed (similar to [8]) were aimed at causing packet collisions between nodes. This would cause greater backoff time which would leave the medium free for longer periods of time when it could otherwise be used. In this way, it was hoped to make the network attack itself by reaching a point where the load was so great that collisions would be unavoidable; sort of a domino effect. This means the jamming node would have to do very little to maintain its effectiveness. It was intuitive that the best way to ensure collisions should be to transmit only high priority (Voice) packets while performing no backoff, $CW_{min} = CW_{max} = 0$, or even a "negative" backoff by additionally setting the AIFSN < 2. It was quickly discovered, however, that the collisions were not happening often enough to create the desired condition. The AIFSN = 0, or 1 was particularly ineffective because the jamming node always took control of the medium well before anyone else, so there was no chance of collision. There was no meaningful effect and the network throughput even increased a little. The no backoff scenario did reduce the throughput by approximately 43%; see Fig.9. Both of these techniques are also highly visible to any node and easy to detect.

The staggered AIFS times – when combined with the backoff timers – create times when several priority levels may have the opportunity to transmit. A Best Effort packet with a backoff timer of 1 will attempt to transmit at the same time as a Video or Voice packet with a backoff timer of 2; see Fig. 2. The next technique attempted to take advantage of this by using the distribution of ACs (35% Voice, 35% Video, 20% Best Effort, and 10% Background) given earlier while still performing no backoff. This was much more effective, dropping the throughput by 62%, to around 600 Kbps. The results for the no backoff scenarios are illustrated in Fig.9.

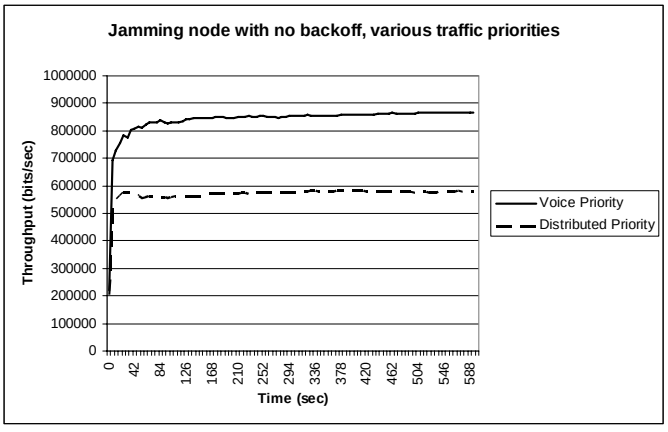

Fig.9: Priority implications of jamming traffic.

The decrease in throughput when lower priority traffic is added to the jammer indicates that the lower priority traffic may be more effective in jamming. Also, due to the relatively small CW for Voice data, the jamming can be just as effective utilizing a backoff which makes it less detectable. Hence, several scenarios were derived to utilize low priority traffic and they were found to be very effective. For these techniques, all traffic generated by the jamming node is of Background priority. The three parameters manipulated are the AIFSN, CW_{min}, and CW_{max}. For brevity, we use the notation [AIFSN, CW_{min}, CW_{max}] to refer to the manipulated parameters.

The following results are similar to those in [8] in that the access time or the CW has been shorted to increase the number of times the jamming node can transmit. In contrast to the 802.11b case, the majority of the traffic on the network is Voice or Video while all of the jamming traffic is Background.

Fig.10 shows the average network throughput with one jamming node sending all Background priority traffic. The key factor here is that the AIFSN has been set to 2 for background instead of the correct 7. It is clear that the smaller AIFSN causes the network to lose throughput. The size of the CW does not have a large effect on the effectiveness of the technique. However, somewhat counter-intuitively, the larger contention windows actually have a slightly lower average throughput toward the end of the simulation. This is due to the buildup of packets due to the larger CW.

The scenario in Fig.10 can be repeated with two misbehaving nodes. The results for the [2, 7, 15] parameters are given later in Fig.13 where results for normal AIFSN and CW sizes are also presented. The result for [2, 7, 15] is an average throughput of about 43,000 bps or 2.8% of normal throughput which is similar to what was seen for 802.11b.

We saw earlier that the misbehaving node was actually transmitting fewer bits than any other nodes in the network. Fig.11 demonstrates that the misbehaving node also has even less throughput than any given normal node. Hence, the misbehaving node is not just flooding the network to decrease overall throughput. Please note that in the subsequent graphs, the bps throughput often have different scales. In Fig.12, the CW parameters of the jamming node are held constant, while the AC or priority loads of the normal nodes are varied. The Normal, Uniform, and Low refer to the value of the parameters and not an underlying probability distribution. Once again, the percentages of AC loads does not impact on the results.

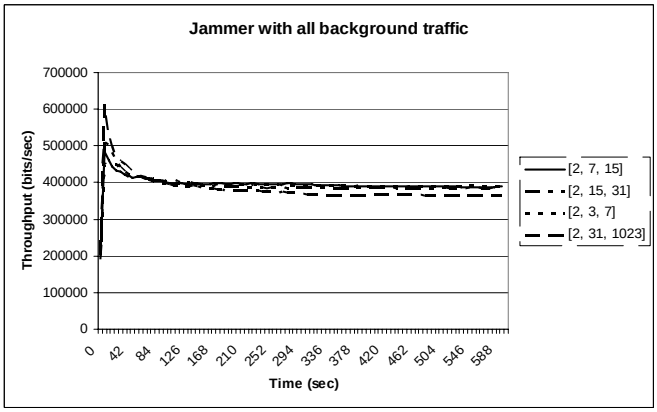

Fig.10: AIFSN = 2, Various contention windows.

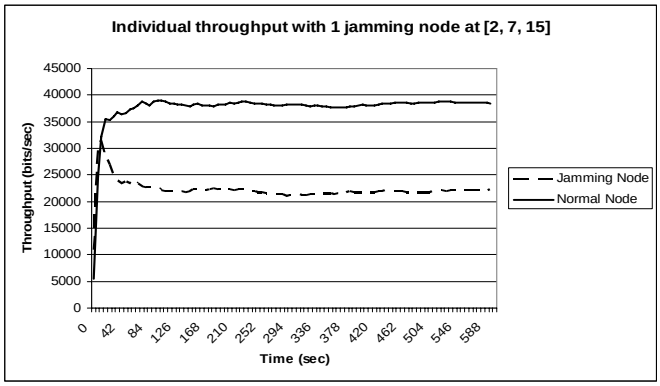

Fig.11: Throughput for misbehaving and normal nodes.

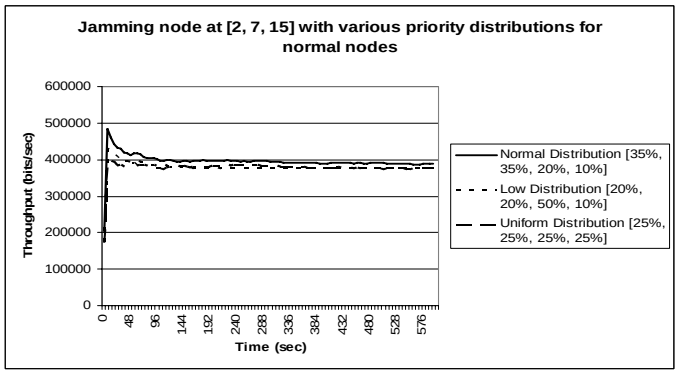

Fig.12: AC percentages for normal nodes.

V. LEGAL JAMMING

We have seen that significant jamming results of 802.11e network can be obtained by transmitting a modest load of 150 byte (average size) packets of Background traffic. The initial intent and result of this approach is similar to that of the "misbehaving node" described previously for 802.11b. Transmitting small packets in quick succession results in a decrease in the total throughput of the network because other stations are not able to transmit. In 802.11b this was accomplished by ignoring the backoff timer and/or contention window. As shown in the previous section, the process is easier in 802.11e due to a side effect of the QoS implementation. We will now show that the process for 802.11e is fundamentally different in that the jamming can be accomplished with the modest load as above but now it is accomplished while satisfying the 802.11e protocol.

In a normal network of nine nodes, the probability of a successful transmission is dependent on the likelihood that any other node attempts to transmit nearly simultaneously. However, in 802.11e each node now has four buffers, one for each AC. This results in internal collisions within a node that can further delay transmission of lower priority messages even when no other node is attempting to send. In the event of collisions within a node, the highest priority packet is transmitted and the others do exponential backoff. Obviously many small packets will increase the likelihood of collisions both within each node as well as between nodes.

In Fig 11, we saw that a single jamming node with the AIFSN = 2 and various CWs was very effective. Fig.13 shows that for two jamming (actually legal) nodes with the legal setting of [7, 31, 1023] provide very effective techniques to

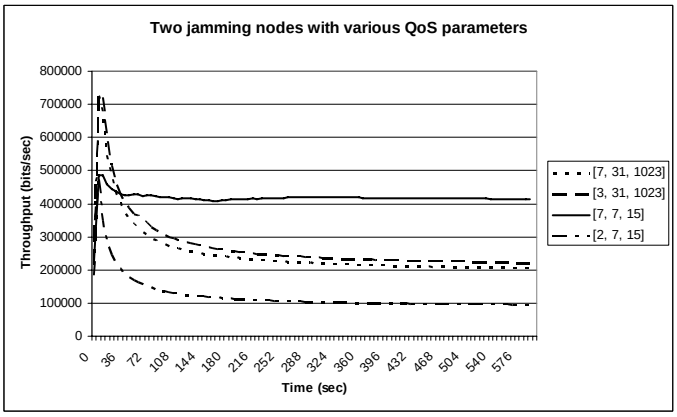

Fig.13: Various QoS parameters.

disrupt the network throughput. Interestingly, the illegal parameter set [7, 7, 15] is 100% **less** effective.

As indicated earlier, the "misbehaving" node was actually transmitting fewer bits than any of the other nodes in the network. Fig.14 demonstrates that the misbehaving nodes (not misbehaving any more) also have even less throughput than any given normal node. Again, the misbehaving node is not just flooding the network to decrease overall throughput.

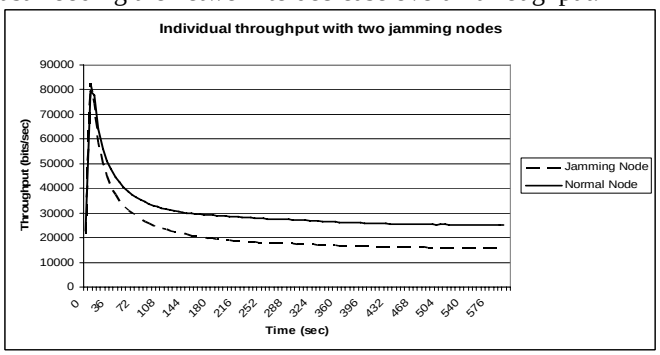

Fig.14: Individual node throughput – legal background.

As seen in Fig.11 for a single jamming node, the different AC load percentages make essentially no difference in network throughput for two "jamming" node. This is shown in Fig.15.

The jamming load has been sent as Background traffic and has been shown to be very effective in decreasing network throughput. One might ask if the same traffic had been sent as Voice, or Video, would it have a similar effect. Intuition would seem to say the "jamming" traffic at Voice or Video would

have a more significant effect. Fig.16 shows that the jamming at Background has the greatest effect with the Video jamming having almost no effect. Best effort is almost the same as Background. Jamming with voice packets decrease the throughput modestly and this appears to be an anomaly (to the counter-intuitive results) that needs further investigating. Fig.16 depicts the situation where two nodes have a modest load of small packets and, in deference to network throughput, transmit at Background priority. One would expect that this would minimize the negative effect on the throughput. However, Fig.16 shows that this is most detrimental to network throughput and if the load were sent at Video AC, there would be almost no effect instead of the 85% decrease experienced. This is totally counter-intuitive and seems to represent a weakness in the 802.11e protocol.

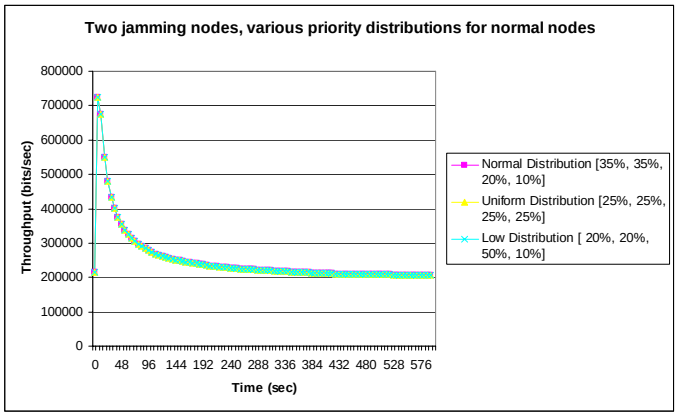

Fig.15: Throughput for priority load percentages.

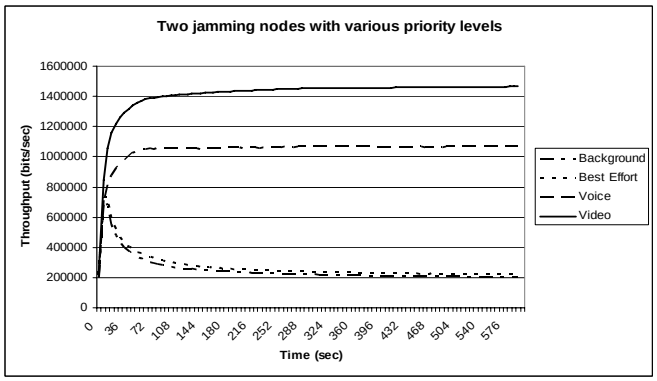

Fig.16: Two jamming nodes – transmitting at each AC.

In this section of the paper, we have shown that successful jamming can be achieved without any violation of the 802.11e protocol. Jamming in this way would be nearly impossible to detect since the jamming station appears to be operating normally and may, in fact, be transmitting normal traffic.

All scenarios show that this modest Background traffic loading is a very effective "jamming" technique in 802.11e. It seems to work regardless of the CW size specified for the jammer and the priority level distribution of the other nodes. Since the jamming node is using normal backoff and the traffic is modest, there is little if anything to indicate a DoS attack.

Our model and load do not satisfy the assumptions in [9] concerning the single AC per node, the uniform packet size (average packet size is very close), and the number of stations are smaller than required for their model. Their single AC per node seems a modeling convenience for the Markov chain approach rather than an intrinsic performance feature. Despite the lack of fit for their model [9], there was a chance the model would predict the decrease in performance that the results here have indicated. The general results from [9] are similar to our results but they do not come close to predicting the dramatic jamming results we have modeled. Another modeling approach [5] provides even less explanation for our results. We do not know of any models for predicting performance of 802.11e that could be applied to our assumptions.

VI. Conclusions

The OPNET simulations show approximately 70% decrease (from 1.5 Mbps to 400 Kbps) in throughput with one node transmitting small packets of Background traffic with a decreased CW. The AIFSN is not changed from the standard setting of 7. This result is almost the same as in 802.11b, but is harder to detect because the node is not ignoring the backoff, merely shortening it, which is already part of the QoS in 802.11e. However, the most stunning results are when nodes transmit small packets of Background traffic. When two nodes are used, it is not even necessary to alter the CW at all to obtain even greater jamming effectiveness. The throughput decreases by 87% (from 1.5 Mbps to 200 Kbps). This is extremely effective as well because neither node involved is violating the protocol rules. These results point to the idea that the important factor here is not utilizing the medium, but loading it with low priority Background traffic. Using two "jamming" nodes generate more traffic. Therefore, it was not necessary to reduce the CW to obtain satisfactory results.

It is extremely counter-intuitive that transmitting large numbers of Background data packets would effectively jam the network. Each normal node in the network is only transmitting 10% Background traffic, so even with the jamming node the offered load on the network for Background traffic represents only 18% of the total offered load.

The effectiveness of this technique could be because the Background traffic the AP is attempting to transmit is colliding with the Voice and Video data from the other nodes, which are at full backoff. The initial transmissions from the normal nodes would collide with the more frequent transmissions from the jamming node causing them to be at full backoff stage (CW = CW_{max}). The average backoff time slot for Voice data in this state is 8. Because the AP is allowed to transmit each AC, one slot sooner than a normal node, the AIFS for Background traffic from the AP would preempt this. So if the jamming node does not preempt or collide with a normal node's transmission, the AP probably will.

Future work includes validating these results in an actual 802.11e network. This work should be done for the 802.11g physical layer. Other techniques, such as the hybrid jamming mentioned in [8], would further reduce any DoS signature.

References

[1] IEEE Std 802.11b-1999/Cor 1-2001 Standard for wireless LAN Medium Access Control (MAC).

[2] IEEE Std. 802.11e-2005 Wireless LAN Medium Access Control Quality of Service Enhancements.

[3] M. Acharya, D. Thuente, "Intelligent Jamming Attacks, Counterattacks and (Counter)^{2}Attacks in 802.11b Wireless Networks", OPNETWORK 2005.

[4] M. Acharya, T. Sharma, D. Thuente, D. Sizemore, "Intelligent Jamming in 802.11b Wireless Networks", OPNETWORK 2004.

[5] M. Kim, J. Ryu, T. Byun , K. Han, "Throughput Analysis of IEEE 802.11e EDCA Protocol", HSNMC 2004.

[6] P. Kyasanur, N. Vaidya, "Detection and Handling of MAC Layer Misbehavior in Wireless Networks", DSN 2003.

[7] M. Raya, J. Hubaux, I. Aad, "DOMINO: A System To Detect Greedy Behavior In IEEE 802.11 Hotspots", 2nd International Conference on Mobile Systems, Applications, and Services, 2004.

[8] D. Thuente, M. Acharya, "Intelligent Jamming in Wireless Networks with Applications to 802.11b and Other Networks", MILCOM 2006.

[9] L. Xiong, G. Mao, "Saturated Throughput Analysis of IEEE 802.11e Using Two-Dimensional Markov Chain Model", Computer Networks: The International Journal of Computer and Telecommunications Networking, Volume 51 , Issue 11, 2007.