Partial Deafness: A Novel Denial-of-Service Attack in 802.11 Networks*

Jihyuk Choi, Jerry T. Chiang, Dongho Kim, and Yih-Chun Hu

University of Illinois at Urbana-Champaign, USA
{jchoi43,chiang2,dkim99,yihchun}@illinois.edu

Abstract. We present a new denial-of-service attack against 802.11 wireless networks. Our attack exploits previously discovered performance degradation in networks with substantial rate diversity. In our attack, the attacker artificially reduces his link quality by not acknowledging receptions (which we call "partial deafness" because an attacker pretends to have not heard some of the transmission), thereby exploiting the retransmission and rate adaptation mechanisms to reduce Medium Access Control (MAC)-layer performance. As compared to previously proposed attacks, the partial deafness attack is particularly strong because the attacker does not necessarily need any advantage over normal users in terms of transmission power, computation resources, or channel condition.

Previous work has shown that time fairness in sharing the wireless medium can improve network throughput. We show that time-based regulation at the data queue of the access point can similarly mitigate the negative impact of a partial deafness attacker.

Keywords: IEEE 802.11 DCF, MAC retransmission, Rate adaptation, Denial of service attack.

1 Introduction

Wireless networks based on the IEEE 802.11 standard [1] are widely deployed today for governmental, commercial, and personal uses. Attacks against the 802.11 standard can cause widespread security exploits ranging from mere inconvenience to privacy breaches and machine compromise. Much attention is dedicated to both possible attacks and their respective solutions. For example, the original security scheme specified by 802.11, the Wired Equivalent Privacy (WEP), is shown to be susceptible to various attacks against both the encryption mechanism [2,3,4] and the authentication scheme [5]. Many protocols are proposed to fix these weaknesses [6,7,8].

Other aspects of the 802.11 are also shown to be susceptible to attacks. For example, the virtual carrier sense mechanism is susceptible to a type of Denial-of-Service (DoS) attack where an attacker repeatedly reserves the channel for

* This material is based upon work partially supported by USARO under Contract No. W-911-NF-0710287 and the NSF under Grant No. CNS-0953600.

S. Jajodia and J. Zhou (Eds.): SecureComm 2010, LNICST 50, pp. 235–252, 2010.

long transmissions, thereby starving other users of any transmission opportunities [5]. Many attacks target the backoff mechanism of the 802.11 standard by not backing off as much as specified by the standard [9,10,11]. Backing off less than specified allows the attackers to obtain more access opportunities, and hence higher throughput, than legitimate users.

Heusse et al. demonstrate that even without any malicious intent or misbehavior, a slow connection can still significantly impact the transfer speed of a fast connection because of the fairness mechanism implemented by the Distributed Coordination Function (DCF) at the Medium Access Control layer (MAC) [12]. In particular, since the IEEE 802.11 DCF seeks to fairly grant access opportunities to each station, each station has an equal opportunity to be the next station to transmit a data packet, thus a fast connection regularly has to wait until a slow connection finishes its reception. This performance anomaly together with excessive channel reservation can be viewed as head-of-queue blocking at the wireless medium since the DCF cannot schedule the next station until the current transmitter is finished.

In this paper, we present *partial deafness attack*, a novel DoS attack that builds of Heusse et al.'s observation. Our attack is based on the realization that most commercial access points are implemented with only a single data queue since the 802.11 standard does not specify or recommend any queuing behavior. Thus, if a transmitted packet is not acknowledged, the packet triggers retransmissions and possible rate adaptation (i.e. slowing the data rate), thereby creating head-of-queue blocking at the access point. The head-of-queue blocking then drastically degrades the performance of the wireless network.

Like other DoS attacks, our attack does not aim to give better performance to the attacker, but to reduce the performance of other users. In our attack, each attacker *artificially worsens* his link quality by intentionally failing to acknowledge packet receptions. Our attack impacts the system in a manner similar to a legitimate user with a slow connection. However, by exploiting the retransmission mechanism specified by the 802.11 standard, the impact of our attack becomes much more devastating, especially to the Transport Control Protocol (TCP) performance of other users.

Our work is novel and interesting for two reasons. First, the attacker can carry out our attack targeting the MAC protocol without modifying the MAC layer; second, our attack can consistently impact the system regardless of the opportunistic nature of the physical layer.

Our proposed attack targets the MAC-layer protocol but does not require the attacker to modify the MAC protocol implementation at his station. For example, an attacker can suppress an acknowledgment by turning off the network interface card any time between the start and completion of packet reception. In contrast to many previously proposed attacks that require substantial modification of the firmware or the hardware and are thus often deemed impractical, our attack can be easily implemented in several ways, including methods that do not directly modify the MAC-layer implementation. For example, in Section 4, we detail our implementation of a partial deafness attacker by enabling and

disabling the acknowledgment function in the driver of a commercial Wireless Local Area Network (WLAN) card. In other words, our attack works even when the attacker abides by the same MAC rules as every other node.

An attacker can simply move farther away from the access point to physically worsen his channel condition and impact other users. However, this approach requires the attacker to find a location such that the channel condition is sufficiently weak to regularly results in retransmission, and yet is not weak enough to result in disassociation. If fading causes the attacker to be disconnected, then the attacker cannot impact other users; on the other hand, if fading improves the attacker's channel condition intermittently, then other users can also experience improved transfer rate intermittently. Our attack suppresses the acknowledgment and thus allows an attacker to be able to consistently worsen his channel condition over time, and cause significant degradation of service to other users.

Since the partial deafness attack relies on head-of-queue blocking at the access point, there are many different methods that can mitigate the attack. We propose implementing time-fairness at the access point instead of relying on a single First-In-First-Out (FIFO) data queue. Our proposed solution can be implemented entirely in software, and does not require any changes to the widely used 802.11 MAC protocol.

The rest of the paper is organized as follows: In Section 2 we review some related work. In Section 3 we detail our attack and analytically show the effect of our attack. We show in Section 4 that our attack is indeed practical and causes severe degradation of network performance. In Section 5 we detail a time-fair mechanism and show that this mechanism mitigates the partial deafness attack. We conclude this paper in Section 6.

2 Related Work

The IEEE 802.11 standard is widely deployed due to the unlicensed spectrum in which it operates and the low cost of client devices and access points. As a result, the security of 802.11 attracts much attention. In particular, most research on MAC security focuses on the requirements of confidentiality and integrity. The original security protocol, WEP, is designed to provide privacy and authenticity of data. However, Fluhrer et al. note that weakness in the encryption algorithm used by WEP can be exploited to allow the discovery of session keys [2]. Numerous related attacks exist in the literature [3,4].

While a cryptographic attack has strong adverse effects on users' privacy and protocol's confidentiality and integrity, our work considers another type of attack where the attacker seeks only to deny service to other users. That is, the attacker aims to reduce a protocol's availability. Specifically, we consider the attacks against the MAC-layer protocol specified in 802.11 rather than the pure resource consumption attacks such as the jamming attack (e.g. jamming attack exploiting clear channel assessment [13]).

Attacks on the 802.11 MAC protocol can exploit management vulnerabilities. Bellardo and Savage implement and demonstrate an attack that targets

the authentication/association scheme of 802.11 [5]. Bellardo and Savage note that the deauthentication and disassociation messages are not encrypted, thus an attacker can easily forge these messages. The attacker can then send the deauthentication message to the access point before client's data is received, or the attacker can send the disassociation message to the client before the client's data is transmitted. Ferreri et al. [14] describe DoS attacks against an access point's association and authentication mechanisms.

Attacks on the 802.11 MAC can also exploit media access vulnerabilities. Bellardo and Savage also note that the 802.11 carrier sense mechanism can be easily exploited. For example, in 802.11 networks, a node can only send data during a certain time period after the channel stops being busy. In particular, if not due to retransmission or fragmentation, a user can only transmit data DCF InterFrame Space (DIFS) after channel is available; otherwise the user can transmit data Short InterFrame Space (SIFS) after, where SIFS $<$ DIFS. A very simple method to deny service is to send a short burst every SIFS. Bellardo and Savage present a more sophisticated scheme exploiting the virtual carrier sense mechanism. The 802.11 standard specifies that the MAC frame header of all packets should contain a *duration* field, which specifies how long others have to wait before transmission is allowed in order to avoid collision. Users update their Network Allocation Vector (NAV) with this duration information and keep quiet for the specified duration. Thus an attacker can repeatedly request long channel occupancy time, thereby starving normal clients of channel occupancy.

The benefit of attacking the duration field rather than sending a short burst every SIFS is the amount of power used to carry out the attack. In the duration field attack, an attacker simply initiates a Request to Send (RTS)/ Clear to Send (CTS) handshake along with the specified duration. The handshake in theory would keep the channel busy for roughly 30 ms. The short burst approach, on the other hand, requires sending a short burst every SIFS, or 10 μs in 802.11b/g networks. Our proposed attack performs even better in terms of power saving for the attackers; in particular, our attack can easily occupy 100 ms of channel time without having to send any messages. Moreover, our attack does not require the attacker to have better service, higher power, or closer distance to the access point. Finally, unlike our attack which works on each access point we tested, the duration field attack does not work in many real systems because most vendors do not implement the 802.11 specification correctly [5].

Heusse et al. point out that when a client uses a lower bit rate than others in a 802.11 network, the performance of all clients is considerably degraded [12]. Tan and Guttag subsequently suggest that time fairness can mitigate this performance anomaly and provide better throughput for the WLAN [15]. In this paper, we present an attacker that exploits the conclusion of Heusse et al. by artificially and intentionally creating rate disparities. We show that access point retransmissions exacerbate the anomaly by creating head-of-queue blocking at the access point's data queue. We then adapt the principle of Tan and Guttag's solution and show how to mitigate our attack by implementing time fairness at the access point's data queue.

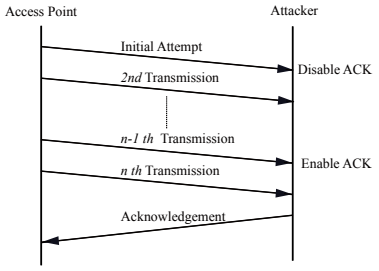


Fig. 1. Partial Deafness Attack

3 The Partial Deafness Attack

3.1 Description

In this section, we present our novel *partial deafness attack*, which exploits the retransmission mechanism of the 802.11 protocol to reduce the bandwidth of non-attacking nodes. In our attack, the attacker, upon receiving a unicast data frame addressed to it, intentionally fails to send a timely acknowledgment for at least a portion of those data frames. Though previous work has suggested denial-of-service attacks against IEEE 802.11, our attack stands out because it substantially reduces the bandwidth available to legitimate nodes without requiring the attacker to have superior connection quality. That is, an attacker with lower transmission power, fewer computation resources, located farther away than a normal client, can still deny service to all the normal clients within the network.

As illustrated in Fig. 1, when a unicast transmission is not acknowledged, an 802.11 station will normally transmit a frame up to seven times before it gives up and discards the frame. An attacker can thus fail to acknowledge the first six transmissions. In addition, senders in 802.11 employ *rate adaptation* (e.g. Auto Rate Fallback (ARF) [16], SampleRate [17]) to maximize the throughput of the channel. When a receiver repeatedly fails to receive transmissions at one bit rate, the sender chooses a lower bit rate in an attempt to successfully deliver the packet. Eventually the sender will choose the lowest possible rate, called the *base rate*, to deliver packets to the attacker.

Since most 802.11 networks are infrastructure networks in which clients connect directly to an access point, and most traffic is directed to or received from an access point, the behavior of an access point plays an important role in the fairness perceived by a station. The 802.11 standard does not specify or recommend any queuing behavior at the access point, so most commercial access points use a single queue. Thus all packets are treated with the same priority and each packet is completed before subsequent packets can be serviced, regardless of the number of retransmissions, or the rate that is selected for those retransmissions. The attacker can thus induce the access point to spend a large amount of time to transmit to the attacker, thereby drastically decreasing the time allocated to the normal clients, and reducing the overall throughput.

3.2 Analysis

We will first analyze the impact of our attack in 802.11b, where the maximum rate is 11 Mbps and the base rate is 1 Mbps. We then use a theoretical analysis to show that rate diversity exacerbates the problem; thus, in commonly deployed 802.11b/g networks, where the maximum and base rates are 54 Mbps and 1 Mbps respectively, are even more susceptible to our attack.

To quantify the degree of imbalance caused by the partial deafness attack, we consider a case in which a normal client and a malicious client share one base station. We call the normal client Alice; the malicious client, Mallory; and base station, Bob. In our example, Alice and Mallory have the same link quality to Bob, so when Mallory is not performing any attack, Bob can send to both Alice and Mallory at 11 Mbps. That is, if Alice and Mallory started User Datagram Protocol (UDP) downloads, they would each receive approximately half of the available bandwidth.

Let us consider the particular rate adaptation algorithm implemented on a Linksys WRT54G access point. Initially, Bob's rate adaptation chooses 11 Mbps for its first three transmissions and 2 Mbps for its last four retransmissions. If Mallory acknowledges after the 3^{rd} transmission, Bob determines that 11 Mbps is too high an initial rate, and will send the subsequent packet at 5.5 Mbps for the first three transmissions and 1 Mbps for the next four retransmissions. If Mallory again acknowledges after the 3^{rd} transmission, Bob determines that 5.5 Mbps is again too high an initial rate, and will send the subsequent packet at 2 Mbps for the first three transmissions and 1 Mbps for the next four retransmissions. If Mallory again acknowledges after the 3^{rd} transmission, Bob will determine that 2 Mbps is still too high and will send all subsequent packets at 1 Mbps.

If Mallory performs the partial deafness attack, and she does not acknowledge receiving a packet until the 7^{th} transmission, Bob would send packets to Mallory at 1 Mbps in the steady state, but to Alice at 11 Mbps. Thus, it would take Bob 11 times longer to send an identical packet to Mallory than to Alice. In other words, if Bob sends an equal number of packets to Alice and Mallory, without considering retransmission, Mallory is already allocated $\frac{11}{12} = 91.7\%$ of the channel occupancy time as opposed to 50% in a time-fair scheme.

We now consider the additional effect of retransmissions. In the Direct Sequence Spread Spectrum (DSSS) mode of 802.11b, the slot time is 20μs, minimum and maximum contention window size are 31 and 1023. Typically 802.11 networks are configured to allow a maximum transmission unit of around 2304 bytes. In 802.11, a station can fragment larger packets into smaller fragments and transmit each fragment separately. In this case, Mallory allows Bob to send each fragment the maximum number of times before Bob gives up on the fragment. Thus each fragment of the packet is transmitted seven times, which is nearly equivalent to transmitting the entire packet seven times. (There are minor differences because of the interframe spacing used between fragments, but seven retransmissions of one large frame should closely approximate seven retransmissions of each of several smaller fragments).

We now quantify Mallory's per-packet channel occupancy time in steady-state. We assume that every time the sender (in this case Bob) wishes to send a packet, the medium is busy, so the 1st transmission experiences backoff. We further assume that once the medium becomes idle, there are no further transmissions on that medium except those initiated by Bob. We will validate the theoretical results here with implementation results in Section 4, which show that these assumptions provide results comparable to those seen in normal access point behaviors. We will consider a single UDP packet containing 1470 bytes of data, which, after UDP- and Internet Protocol (IP)-layer headers, comes to 1498 bytes. The addition of MAC-layer headers brings the total to 1534 bytes.

If Alice and Mallory both acknowledge reception of a packet by the 3rd transmission, the steady-state data rate is 11 Mbps. In this case, the 1st transmission takes about 1571.6 μs in expectation: 50 μs for DIFS, 310 μs of expected backoff, 96 μs of preamble, and 1115.6 μs of data. Bob would expect an acknowledgment within 126 μs, which represents the sum of: the SIFS that Mallory must wait following reception, the maximum propagation delay between Mallory and Bob, which is defined in 802.11 to be one slot time, and the delay that 802.11 allows between when the radio frequency energy starts impinging on the receiver until that receiver starts receiving a message, which is defined to be the length of the preamble. In expectation, a failed 1st transmission would therefore be detected 1697.6 μs after the medium becomes idle. When the 1st transmission is successful, Mallory waits SIFS and transmits a preamble and a 12 byte acknowledgment at 2 Mbps, which gives an expected time of 1725.6 μs from when the medium is idle until the transmission is received. (We assume the propagation time is negligible; the 20 μs slot time of 802.11 is sufficient for a 6 km transmission, which is well in excess of typical 802.11 transmission distances). In further retransmissions, the one thing that changes is the expected backoff value, which increases from 310 μs to 630 μs to 1270 μs within these first three retransmissions. Also, Bob will not wait DIFS when Bob does not receive an acknowledgment. Thus success after three retransmissions takes $1697.6 + (1647.6 + 320) + (1675.6 + 320 + 640) = 6300.8$ μs. If Mallory forces three retransmissions for each packet while Alice acknowledges every 1st transmission, then Mallory will capture $\frac{6300.8}{6300.8+1725.6} = 78.5\%$ of the channel occupancy time.

When Bob must regularly transmit each packet at least four times in order to reach Mallory, Bob sends every packet to Mallory at 1 Mbps. Thus each data transmission takes 12272 μs for data alone, which, after adding backoff, preamble, and header for the 1st transmission takes 12678 μs. The acknowledgment times out after the same 126 μs, giving a failure time for the 1st transmission of 12804 μs. Thereafter, each failure takes the same amount of time after adjustment for backoff, and when the acknowledgment finally comes, it is transmitted at 1 Mbps, so seven retransmissions takes $50 + 12678 * 7 + 126 * 6 +$ backoff increases $+ 202$ (μs), where 50 μs is DIFS, 12678 μs is the time that each packet transmission takes, 126 μs is the time to detect that an acknowledgment is not forthcoming, and 202 μs is the time to finish receiving an acknowledgment. The total additional backoff for seven retransmissions is 28160 μs in expectation,

so the total transmission time is 117914 μs. If Mallory forces six retransmissions (for a total of seven transmissions) for each packet while Alice acknowledges every 1st transmission, then Mallory will capture $\frac{117914}{117914+1725.6} = 98.6\%$ of the channel occupancy time.

Finally, we argue that rate diversification exacerbates the partial deafness attack. In the same scenario, when Alice uses a 54 Mbps link in a 802.11b/g network, Mallory's transmissions take the same amount of time, but Alice's transmissions are now much faster. The DIFS and backoff take 360 μs as before (because it is a mixed-mode 802.11b/g access point), 802.11g does not require a preamble, and Alice's data transmission is now 227.3 μs, for a forward transmission time of 587.3 μs; after a 10 μs 802.11g SIFS and a 30 μs 802.11g acknowledgment, each Alice's packets take 627.3 μs in expectation. Thus Alice's channel occupancy time drops further to 0.53%.

4 Implementation and Evaluation of the Partial Deafness Attack

In this section, we detail our implementation of a partial deafness attacker and observe that the attack does in fact impact the data rate greatly.

4.1 Implementation

We implemented a partial deafness attacker to see the effect of the attack on an 802.11 network. Our implementation uses commercial off-the-shelf 802.11 Network Interface Cards (NICs). Most commodity 802.11 NICs generate and send acknowledgment frames automatically in firmware whenever a packet is received, because of the hard real-time deadlines on generating acknowledgments. The partial deafness attack can then be implemented by building custom hardware, modifying the firmware to defer acknowledgments, or turning off the network interface card any time between the start and completion of packet reception.

In order to simplify the task of deferring packet acknowledgments, we choose to modify the MadWifi driver, which is a Linux kernel device driver for Atheros-based WLAN devices. The Atheros chipset does not load a firmware onto the card, but instead relies on a Hardware Abstraction Layer (HAL) module that is part of the driver. The HAL module defines the interface between the hardware and other software in the device driver to manage many of the chip-specific operations and to enforce any relevant regulations.

We modified MadWifi to control a particular register in the HAL module that allows us to enable and disable packet acknowledgments. As illustrated in Fig. 1, we suppressed acknowledgements from the first $n - 1^{\text{th}}$ transmissions by switching the HAL register.

Our evaluation network consists of a traffic source connected to an IEEE 802.11b/g access point. A normal user and an attacker use 802.11 to connect to the access point. This topology is illustrated in Fig. 2. We use two different kinds of access points in our experiment. When we do not need to modify the

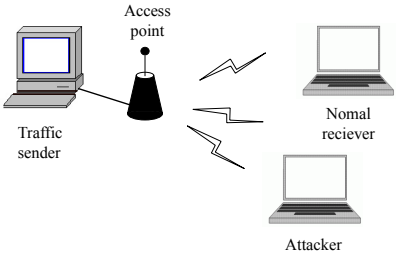


Fig. 2. Network Topology

access point queuing algorithms, we use commercial off-the-shelf access points such as Linksys WRT54G, which uses the Broadcom BCM5352EKPB chipset and supports 802.11b/g mixed mode, because it shows how the rate adaption is practically implemented in real 802.11 system. When we do need to modify the access point queuing algorithms, we use HostAP on a Pentium-III 1 GHz laptop running Linux 2.6.24 because we cannot control queuing behavior in the commercial products to which we have access. The Pentium-III laptop has an Ethernet interface and an Atheros 802.11a/b/g card. We use MadWifi and configure the Atheros NIC to operate in 802.11 master mode. We then use kernel-level bridging to bridge between the 802.11 network interface card and the Ethernet network interface card. For traffic generation, we use iperf; the traffic source generates traffic as an iperf client, which was then sunk at iperf servers running on the normal user and the attacker. We collect our data through an additional machine (not shown in Fig. 2), which captures all 802.11 frames sent on the network.

4.2 Evaluation

Maximum Throughput of Attacker. In order to determine the bit rate that an attacker needs to send to saturate the channel, we first examine the maximum throughput of the attacker using 802.11b when the attacker is the only user of the access point. We perform these measurements and theoretical analysis using UDP because UDP is a non-conforming load and will allow us to set our load regardless of the route's capability to handle that load. When Mallory forces Bob to transmit each packet n times, we compute the amount of time required per packet as described in Section 3; we then translate this into an application-layer rate and present it in Table 1.

As described in Section 3, the rate adaptation mechanism at the access point selects an 11 Mbps rate for users that acknowledge at least once every 3 transmissions and selects a 1 Mbps rate for users that acknowledge less frequently than every 3 transmissions. This contributes to the sharp reduction in maximum throughput between a user who acknowledges every 3 packets and a user who acknowledges every 4 packets.

We then implemented the partial deafness attacker that requires 1 to 7 transmissions before it will send an acknowledgment. We could not consistently

require 2 transmissions because the driver we used to enable and disable acknowledgments could not consistently set the register within the real-time requirement between the first and the second transmissions. We ran this attacker both in an outdoor environment without measurable 802.11 interference and in an indoor environment where the 802.11 interference was uncontrolled. Some experimental results are greater than the calculated theoretical values because the access point, in violation of the specification, interleaves a beacon transmission between retransmissions of the original data packet. Because beacons are broadcast, and because broadcast messages are always considered successful, they reset the contention window size to minimum without resetting the retry count. Appendix A provides further details. Our results show that a partial deafness attacker receiving about 115 kbps of traffic can exhaust the entire forwarding capability of an access point.

Impact on UDP victim. We consider the impact on the throughput of a normal client that uses UDP against a partial deafness attacker that only acknowledges the 7^{th} transmission of each packet. Theoretically, if the access point receives α packets destined to the normal user for every packet destined to the attacker, then we would expect that the normal user would get a $\frac{\alpha}{1+\alpha}$ share of the overall throughput, since the access point treats all packets equally.

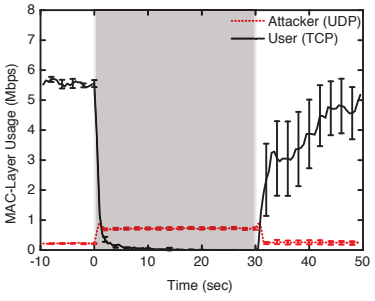
To test this hypothesis, we gave the attacker a UDP source rate of 200 kbps, which is sufficient to saturate the access point's wireless link under the partial deafness attack; and the normal user, a UDP source rate of 100, 200, then 400 kbps. The resulting throughput is shown in Table 2. As expected, the ratio of throughputs is equal to the ratio of the UDP source rates.

Table 1. Maximum UDP throughput of an attacker. n is the number of transmissions required before the attacker sends an acknowledgment; this table shows results in a theoretical analysis as described in Section 3 and an actual outdoor/indoor experiment without/with any detectable 802.11 interference.

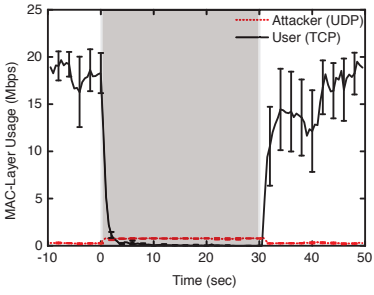
n	Theoretical	Outdoor	Indoor
1	6814.9 (kbps)	6049.0 (kbps)	5782.0 (kbps)
2	3184.2 (kbps)	N/A	N/A
3	1866.4 (kbps)	1563.0 (kbps)	1282.0 (kbps)
4	214.4 (kbps)	209.1 (kbps)	193.3 (kbps)
5	162.3 (kbps)	163.2 (kbps)	159.4 (kbps)
6	123.5 (kbps)	128.8 (kbps)	123.2 (kbps)
7	99.7 (kbps)	115.0 (kbps)	114.0 (kbps)

Table 2. UDP throughputs under partial deafness attack. Attacker's source rate is 200 kbps. Results are averaged over 20 runs.

Normal user's source rate	Normal user's throughput	Attacker's throughput
100 (kbps)	55.7 (kbps)	112.0 (kbps)
200 (kbps)	111.8 (kbps)	111.7 (kbps)
400 (kbps)	219.1 (kbps)	109.3 (kbps)



(a) Impact on 802.11b normal user



(b) Impact on 802.11g normal user

Fig. 3. MAC-layer utilization by TCP under the partial deafness attacker. The shaded region (0-30 sec) shows the time of attack; results are averaged over 20 runs, with the error bars (95% confidence interval).

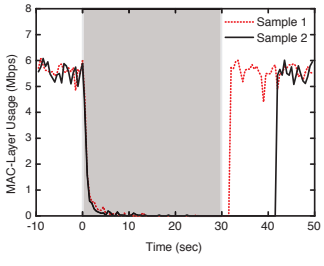


Fig. 4. The differences of TCP recovery time. The shaded region (0-30 sec) shows the time of attack.

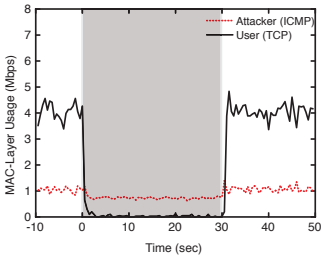


Fig. 5. Partial deafness attack using ICMP ping. The shaded region (0-30 sec) shows the time of attack.

Impact on TCP victim. We now consider a normal TCP user competing for bandwidth against a partial deafness attacker. The attacker again uses a UDP source rate of 200 kbps. To show the impact of the attack, we allow TCP to warm up for a period of time before the attack starts; then perform the attack for a period of time, and finally turn off the attack and allow TCP to return to its steady-state behavior. Because we are interested in how nodes share the available bandwidth on the wireless link, we measure MAC-layer bandwidth usage, counting each retransmission as additional channel usage. As shown previously, each transmission to the attacker theoretically takes around 118 ms. We thus quantized each protocol's usage into 500 ms slots so that the normal user has a chance to receive data in each slot, and each slot conveys the granularity of MAC-layer usage. We plotted the MAC-layer usage over time for each scenario. Because we allow a warm-up and cool-down period where the attacker does not perform the partial deafness attack, each plot includes a shaded box covering the 30-second time interval (from 0 to 30) during which the attack took place.

Fig. 3(a) shows the MAC-layer usage when a partial deafness attacker competes against a normal user's TCP flow when both clients use 802.11b. As shown

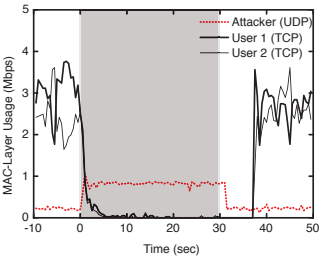


Fig. 6. Partial deafness attack on the network with two 802.11b normal users. The shaded region (0-30 sec) shows the time of attack.

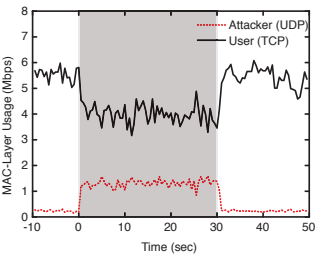


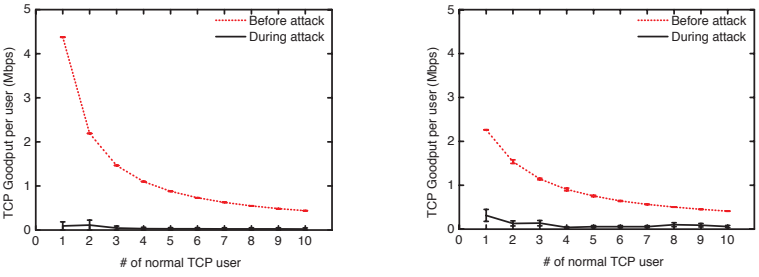
Fig. 7. Partial deafness attack on an access point with fixed rate, 11 Mbps. The shaded region (0-30 sec) shows the time of attack.

in Table 1, a UDP attacker only needs to transmit 115 kbps in order to saturate the link and cause congestion; by allowing the attacker to send 200 kbps traffic would cause the attacker to experience a 43% loss rate without considering a sharing normal user. When a normal TCP user shares the channel with the attacker, the access point treats and drops an equal fraction of UDP and TCP packets, hence the TCP user would experience similar loss rate as the attacker. That is, the normal TCP user would experience at least a 43% loss rate; since TCP is a conforming transport layer protocol, such a high loss rate causes repeated TCP time-out and results in minimal throughput for the normal user, as shown in Fig. 3(a). We observe that TCP has substantial variance in the MAC layer usage during recovery (Fig. 3(a)); to show the cause of this large variance, we plot two sample runs in Fig. 4 and show that the TCP flow in each sample run recovers at substantially different time.

We examined the impact of a partial deafness attacker in the scenario where a normal user connects to the access point using the 802.11g standard. The normal user enjoys a faster connection when the attacker is silent; however, when the attacker carries out the partial deafness attack, the transfer speed of the normal 802.11g user is not significantly faster than that of a normal 802.11b user. This result is consistent with our analysis of rate diversity in a 802.11b/g network at the end of Section 3.

Partial deafness can even be carried out by an unauthenticated station when an access point uses a captive portal to authenticate end points. To attack such an access point, the attacker guests traffic to itself by sending Internet Control Message Protocol (ICMP) ping messages to the captive portal. Fig. 5 shows the impact of the data rate of a normal user when an attacker performs a flood ping (using the '-f' option) where each ping packet contains 1470 bytes of data. Our results shows that an attacker can deny an access point's service, even if the access point uses a captive portal to authenticate users.

The partial deafness attack creates head-of-queue blocking by using retransmission and rate adaptation; thus, a normal user will experience an even higher loss rate when other normal users are also present. This is intuitive since all



(a) UDP attacker (b) TCP attacker

Fig. 8. ns-2 simulation of the partial deafness attack on a network with multiple 802.11b normal users; results are averaged over 20 runs, with the error bars showing 95% confidence interval

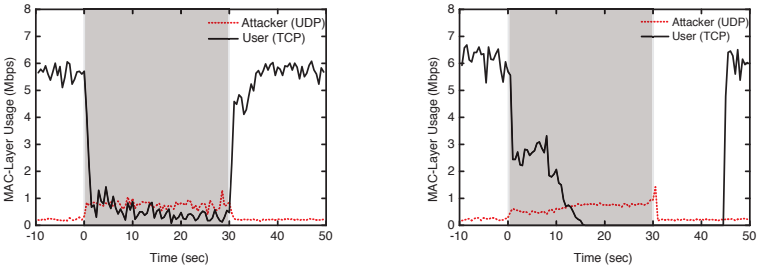
users are going to compete for the limited amount of remaining bandwidth. We performed our partial deafness attack in a network with 2 normal users, and show our results in Fig. 6.

We also performed an ns-2 simulation on the impact of the partial deafness attack in a network with 1 to 10 normal users in addition to the attacker. In our simulation, all users (normal and attacker) are located on a circle 1 m away from the access point. The normal users and the attacker are given identical properties (such as signal and noise power levels), except the acknowledgment policy. That is, the attacker is identical to a normal user except he does not acknowledge receiving a packet until the 7^{th} transmission.

We present our simulation results in Fig. 8. Fig. 8(a) and Fig. 8(b) show the effectiveness of the partial deafness attack when the attacker uses UDP with source rate of 200 kbps and TCP respectively. In both cases, we see the goodput per normal user during attack is minuscule compared to the fair goodput each normal user enjoys without the attack.

The partial deafness attack works by exploiting the retransmission mechanism specified by 802.11 and the rate adaptation implemented at an access point. We thus examined the effectiveness of the partial deafness attack in the scenario where the access point does not support rate diversity. Since a fast connection is impacted by the slow connection partially due to the transfer speed, we expect the impact of partial deafness attack to be alleviated in the case where the access point does not provide rate adaptation. We show our result in Fig. 7.

We examined the effectiveness of the partial deafness attack on two other access points that use different chipsets from that of Linksys WRT54G. Specifically, we examined a Linksys WRT54GC, and a Trendnet TEW-432BRP access points. We present our results in Fig. 9. We observe that both access points are also susceptible to the partial deafness attack. Even though rate adaptation mechanisms of these two access points are different from that of Linksys WRT54G, the partial deafness attack still makes the attacker's traffic use the base rate during attack period. For the Linksys WRT54GC, each packet is



(a) Linksys WRT54GC (b) Trendnet TEW-432BRP

Fig. 9. MAC-layer utilization by TCP under the partial deafness attacker. The shaded region (0-30 sec) shows the time of attack.

retransmitted only 4 times (we discuss this behavior in Appendix A.3). The rate adaptation mechanism in Trendnet TEW-432BRP decreases the rate slowly as compared to the Linksys WRT54G. This difference results in slower performance degradation, as shown in Fig. 9(b).

5 Countermeasure

In this section, we propose a countermeasure that mitigates the partial deafness attack. The partial deafness attack is based on head-of-queue blocking at the access point that results in starvation of normal users. Thus our intuition for mitigating the attack is to use time fairness to prevent starvation. Time fairness has also been suggested in previous work [15] to increase throughput in a network with rate diversity.

Time fairness can be enforced at the access point by implementing a Time-Based Regulator (TBR) that times each transmission: if user A is allocated time duration t_n in the n^{th} round, then all other users are allocated the same time duration.

We implemented a TBR on HostAP as described in Section 4.1. In particular, we implemented a priority queue at the access point that allows us to select the next client to serve. We also emulated the rate adaptation of the Linksys WRT54G access point in order to obtain consistent comparisons of the data rates between our attack scenarios and our mitigation implementation.

Table 3. UDP throughput of normal user and partial deafness attacker with Time-Based Regulator (TBR). The source rate of attacker and normal user is 11 Mbps. Results are averaged over 20 runs.

	Attacker	Normal user
Normal user only		6.07 (Mbps)
Without TBR	110.9 (kbps)	107.9 (kbps)
With TBR	52.5 (kbps)	2.93 (Mbps)

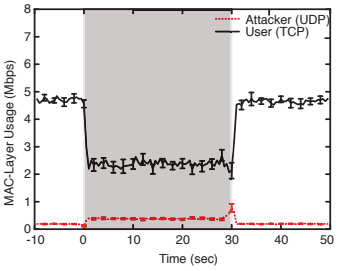


Fig. 10. The TCP user's MAC-layer channel utilization with the countermeasures. The shaded region (0-30 sec) shows the time of attack; results are averaged over 20 runs, with the error bars (95% confidence interval).

We first consider the case where a normal UDP user shares the wireless link with a partial deafness attacker. We gave both the partial deafness attacker and the normal user a UDP source rate of 11 Mbps. The partial deafness attacker is configured to only acknowledge the 7^{th} transmission of every packet. The resulting throughput is shown in Table 3. When there is no attacker, the user can receive 6.07 Mbps of traffic, which is consistent with our previous result in Table 1. Moreover, when the attacker is present, the user still enjoys almost half of this rate, at 2.93 Mbps, which shows a significant improvement over using access opportunity fairness.

We applied a TBR to a TCP user in the presence of a partial deafness attacker who uses UDP at the transport layer. Fig. 10 shows that a TBR allows the user to obtain significantly better service when under attack. In particular, the TCP user ceases to experience heavy packet losses when a TBR is deployed at the access point.

Time fairness can be implemented with 802.11e by choosing appropriate traffic category for each node according to their fair share of channel occupancy time [15]. However, 802.11e itself (i.e. 802.11e without TBR) might not be effective as a countermeasure since 802.11e specifies only four traffic categories (i.e. four queues). As multiple partial deafness attackers can connect to a single access point, the attackers can collectively block all four queues used by 802.11e.

6 Conclusions

In this paper, we presented a denial-of-service attack, called *partial deafness*, against current IEEE 802.11 wireless networks. Our attack targets the 802.11 MAC protocol without modifying the MAC-layer implementation. Furthermore, our attack does not require the attacker to have better resources than a normal user; the attacker can have lower signal strength, slower computation, and be farther from the base station and still negatively impact the normal users. We showed that our attack substantially degrades the performance of normal users that use UDP and can almost completely deny service to users using TCP.

We then proposed and evaluated a countermeasure based on time fairness that mitigates the partial deafness attack. We use time-based regulation to ensure that each client gets an equal fraction of the service provided by the access point. We experimentally showed that this mechanism restores a reasonable level of performance for normal users, whether they use UDP or TCP, when an attacker performs the partial deafness attack.

References

1. IEEE Std. 802.11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications (2007)
2. Fluhrer, S., Mantin, I., Shamir, A.: Weaknesses in the key scheduling algorithm of RC4. In: Vaudenay, S., Youssef, A.M. (eds.) SAC 2001. LNCS, vol. 2259, pp. 1–24. Springer, Heidelberg (2001)
3. Stubblefield, A., Ioannidis, J., Rubin, A.D.: A key recovery attack on the 802.11b wired equivalent privacy protocol (WEP). ACM Transactions on Information and System Security 7(2), 319–332 (2004)
4. Bittau, A., Handley, M., Lackey, J.: The final nail in WEP's coffin. In: 27th IEEE Symposium on Security and Privacy, pp. 386–400. IEEE Computer Society, Los Alamitos (2006)
5. Bellardo, J., Savage, S.: 802.11 denial-of-service attacks: Real vulnerabilities and practical solutions. In: 12th USENIX Security Symposium, pp. 15–27. USENIX Association, Berkeley (2003)
6. Wi-Fi Alliance: Wi-Fi Protected Access: Strong, standards-based, interoperable security for today's Wi-Fi networks (2003)
7. IEEE Std. 802.11i: Medium Access Control (MAC) Security Enhancements (2004)
8. IEEE Std. 802.1X: Port-Based Network Access Control (2004)
9. Kyasanur, P., Vaidya, N.H.: Selfish MAC layer misbehavior in wireless networks. IEEE Transactions on Mobile Computing 4(5), 502–516 (2005)
10. Cardenas, A.A., Radosavac, S., Baras, J.S.: Performance comparison of detection schemes for MAC layer misbehavior. In: 26th IEEE Conference on Computer Communications, pp. 1496–1504. IEEE Communication Society, Piscataway (2007)
11. Raya, M., Hubaux, J.P., Aad, I.: DOMINO: A system to detect greedy behavior in IEEE 802.11 hotspots. In: 2nd International Conference on Mobile Systems, Applications, and Services, pp. 84–97. ACM, New York (2004)
12. Heusse, M., Rousseau, F., Berger-Sabbatel, G., Duda, A.: Performance anomaly of 802.11b. In: 22nd IEEE Conference on Computer Communications, pp. 836–843. IEEE Communication Society, Piscataway (2003)
13. Denial of service vulnerability in IEEE 802.11 wireless devices, `http://www.auscert.org.au/render.html?it=4091`
14. Ferreri, F., Bernaschi, M., Valcamonici, L.: Access points vulnerabilities to DoS attacks in 802.11 networks. In: IEEE Wireless Communications and Networking Conference, pp. 634–638. IEEE Communication Society, Piscataway (2004)
15. Tan, G., Guttag, J.: Time-based fairness improves performance in multi-rate WLANs. In: USENIX Annual Technical Conference, pp. 269–282. USENIX Association, Berkeley (2004)
16. Kamerman, A., Monteban, L.: WaveLAN-II: A high-performance wireless lan for the unlicensed band. Bell Labs Technical Journal 2(3), 118–133 (1997)

17. Bicket, J.C.: Bit-rate Selection in Wireless Networks. Master's thesis, Massachusetts Institute of Technology (2005)
18. Han, B., Schulman, A., Gringoli, F., Spring, N., Bhattacharjee, B., Nava, L., Ji, L., Lee, S., Miller, R.: Maranello: Practical partial packet recovery for 802.11. In: 7th USENIX Symposium on Networked Systems Design and Implementation. USENIX Association, Berkeley (2010)

A Time Distribution of Beacon-Induced Backoff

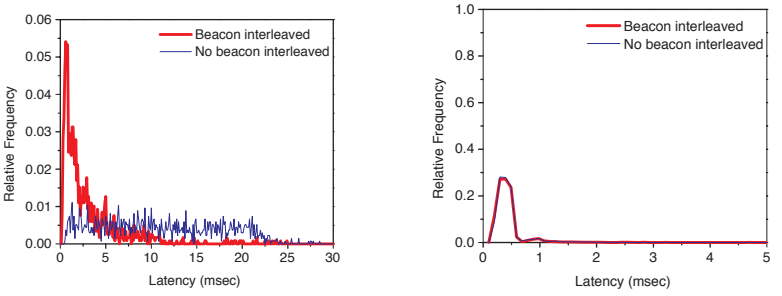
Each access point periodically broadcasts beacons. Since beacons are broadcast messages, they are not acknowledged, so the 802.11 standard considers all beacon transmissions successful. However, when beacons are transmitted between retransmissions, the perceived success from the broadcast causes the access point to choose a contention window on the interval between $[0, \mathrm{CW}_{\min}]$. Furthermore, since the packet waiting for retransmission has not yet been acknowledged, the access point does not reset the retry limit counter. This creates significant discrepancies in the backoff process between what the standard specifies and what actually happens using commercial products.

To demonstrate the discrepancies caused by the periodic beacons, we examine the latency between the 6^{th} and 7^{th} transmission. The 802.11 standard specifies that the 7^{th} transmission wait Short InterFrame Space (SIFS) (10 μs) and then backoff with a value uniformly distributed over [0, CW[7]]. However, if the contention window were reset between the m^{th} and the $(m+1)^{\text{st}}$ transmission, the resulting backoff between the 6^{th} and 7^{th} transmission would be off by a factor of 2^{m-1}. (We use $m-1$ instead of m because CW[6] = CW[7] in 802.11b.) Therefore, given the beacons are transmitted periodically, we should expect the latency to be distributed geometrically.

A.1 Broadcom Chipset

We examined a Linksys WRT54G (ver. 5) using a Broadcom chipset. By default, this access point sends a beacon message every 100 ms. However, as shown in Section 3, the total time required to send 7 transmissions almost always takes longer than 100 ms. Thus, we change the beacon interval to 200 ms in order to demonstrate the effect of the beacon messages.

Fig. 11(a) shows a histogram of the latency between the 6^{th} and 7^{th} packet transmission with each bin size 100 μs, equaling 5 slot time. We categorized transmissions into two sets: one set contains all the transmissions where a beacon packet had been interleaved between the 1^{st} transmission of this packet and the 7^{th}; the other set contains all the transmissions for which no beacon packet had been interleaved between the 1^{st} transmission of this packet and the 7^{th}. The thin line shows the latency of the second set; that is, when no beacon has been interleaved. In the non-interleaved case, the latency is uniformly distributed, as would be expected from reading the 802.11 standard. The bold line shows our observation of latency from the first set; that is, for packets into which beacons have been interleaved. In this case, the latency is exponentially/geometrically



(a) Linksys WRT54G (b) MadWifi

Fig. 11. Latency between the 2nd to last and last retransmissions of the same packet

distributed, which shows that beacons are interleaved and this interleaving does affect the backoff values chosen.

A.2 MadWifi Driver

As described in Section 4.1, the Hardware Abstraction Layer (HAL) module operates between the hardware and the device driver and is implemented to manage many of the chip-specific operations. The HAL module is distributed with the driver. Thus, the same Atheros Network Interface Card (NIC) may exhibit different behaviors when using different drivers that contain different HALs.

With the same scenario as described in previous section, we tested the MadWifi driver and the Atheros NIC by using HostAP. We observed that the MadWifi driver does not increase its contention window when retransmitting packets as shown in Fig. 11(b). There had been suspicions that MadWifi driver is not backing off properly [18]; moreover, when we used Windows and a Windows driver with the same Atheros NIC in an ad-hoc connection, we did not observe the improper backoff behavior. We thus conclude that MadWifi driver does not perform exponential backoff properly.

A.3 Marvell ARM914 Chipset

We also tested the Linksys WRT54GC access point, which uses the Marvell ARM914 chipset.We found that the maximum number of retransmissions was 4 instead of 7. The 802.11 standard specifies that packets with payload longer than Request to Send (RTS) threshold are transmitted up to the long retry count of 4, and with payload shorter than RTS threshold are transmitted up to the short retry count of 7. Most access points, including the Linksys WRT54GC, set the RTS threshold so that all packets are sent without RTS/Clear to Send (CTS), and thus each packet should be retransmitted up to 7 times. We thus conclude that the WRT54GC improperly set the short retry count to 4.