WiFi Says "Hi!" Back to Strangers!

Ali Abedi
University of Waterloo
ali.abedi@uwaterloo.ca

Omid Abari
UCLA
omid@cs.ucla.edu

ABSTRACT

WiFi networks employ authentication and encryption mechanisms to protect the network from being accessed by unauthorized devices. Therefore, WiFi communication should be possible only between devices inside the same network. However, we have found that all existing WiFi devices send back acknowledgments (ACK) to even fake packets received from WiFi devices outside of their network. We call this behavior *Polite WiFi* since WiFi devices respond to all packets even those coming from strangers!

In this paper, we discover the Polite WiFi behavior for the first time. We also examine this behavior on over 5,000 WiFi devices from 186 vendors. We find that all existing WiFi devices respond to fake packets transmitted to them. We believe this behavior creates many threats as well as opportunities. For example, one can couple this behavior with WiFi sensing and localization techniques to create a new class of security threats. In particular, by continuously sending fake frames to a target device, and measuring the properties of the ACK signal, one can extract sensitive personal information. Similarly, an attacker may use this behavior to quickly drain the battery of WiFi devices by continuously sending fake packets to them. Despite these threats, we also believe that the Polite WiFi behavior can open up new opportunities to WiFi sensing applications by making them more practical and easier to deploy.

CCS CONCEPTS

• **Security and privacy** → **Network security**; *Privacy-preserving protocols*; *Privacy protections*; • **Networks** → **Wireless access points, base stations and infrastructure**; **Mobile and wireless security**; **Wireless local area networks**; *Network security*; *Network privacy and anonymity*; • **Hardware** → **Wireless devices**; *Sensor devices and platforms*; • **Computing methodologies** → Machine learning.

ACM Reference Format:
Ali Abedi and Omid Abari. 2020. WiFi Says "Hi!" Back to Strangers!. In *Proceedings of the 19th ACM Workshop on Hot Topics in Networks (HotNets '20), November 4–6, 2020, Virtual Event, USA.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3422604.3425951

1 INTRODUCTION

In a WiFi network, when a device sends a frame to another device, the receiving device sends an acknowledgement back to the transmitter. This mechanism is deployed to deal with error prone wireless channels and to handle retransmissions in the physical and MAC layer. In particular, upon receiving a frame, the receiver calculates the cyclic redundancy check (CRC) of the frame to detect possible errors. If the frame passes CRC, then the receiver sends an Acknowledgment (ACK) to the transmitter to notify the correct reception of the frame.

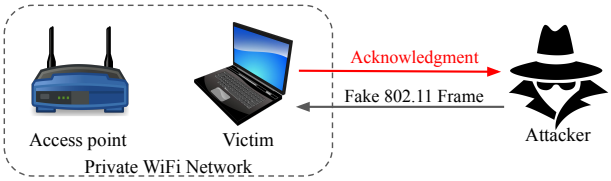


Figure 1: WiFi devices send an ACK for any frame they receive without checking if the frame is valid.

Since most networks use security protocols (such as WPA2) to prevent unauthorized devices from joining, one may assume that a WiFi device only acknowledges frames received from the associated access point or other devices in the same network. However, surprisingly, we have found that all today's WiFi devices acknowledge **any** frame they receive as long as the destination address matches their MAC address. The physical layer acknowledges all frames even those without any valid payload, although higher layers eventually discard the fake packet. Consider a scenario where a client device is connected to an access point, as shown in Figure 1. This is a private network secured by protocols such as WPA2. We have found that if an attacker sends a fake and unencrypted 802.11 frame to the client device (labeled as victim), the client device sends back an acknowledgment! We call this behavior Polite WiFi because WiFi devices respond to any stranger with an acknowledgement.

The Polite WiFi behavior creates many threats. For example, an attacker can send back-to-back fake frames to a victim device. Then, it can analyze the signal of the received ACKs to infer some personal information about the victim. Recent studies have shown that by monitoring the WiFi signal, one can infer some information about the environment such as localization [21, 23], gesture recognition [28, 30], breathing rate estimation [18, 26], and keystroke inference [16]. Hence, the attacker can exploit these systems and use the signal of ACKs to infer some personal information. Another possible threat is the battery-drain attack. An attacker can force a victim to continuously transmit ACKs by bombarding the victim with fake packets. This would drain victim's battery very quickly. Our experiments show that this attack can increase the power consumption of a low-power IoT device by an order of 35x. This can be problematic for many sensitive sensor devices and medical devices.

In this paper, we study the Polite WiFi behavior in more detail on over 5000 WiFi devices. We also explore if this behavior can

HotNets '20, November 4–6, 2020, Virtual Event, USA

ACM ISBN 978-1-4503-8145-1/20/11…$15.00
https://doi.org/10.1145/3422604.3425951

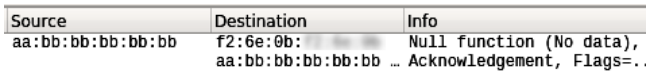


Figure 2: Frames exchanged between attacker and victim

be avoided, and why today's WiFi devices have such behavior. Moreover, we examine the possibility of a few attacks using Polite WiFi. Finally, as opposed to creating threats, we also show how it can be used as an opportunity to help WiFi sensing techniques to be more practical.

The main contributions of this work are:

- We find that WiFi devices respond to fake 802.11 frames by acknowledging the reception of these frames. We have tested over 5,000 WiFi access points and client devices and have found that all of them are susceptible to the Polite WiFi behavior.
- We demonstrate examples of how Polite WiFi can be utilized in conjunction with WiFi sensing techniques to create new threats. Specifically, we show how Polite WiFi makes existing keystroke inference systems significantly more dangerous. In addition, we show that battery-operated WiFi devices can be attacked by forcing them to transmit ACKs which significantly increases their power consummation by 35x.
- We demonstrate how Polite WiFi creates new opportunities for WiFi sensing techniques by making them more practical. We show that sensing techniques can be implemented by making software modifications on only one WiFi device, instead of existing two-device implementations.

2 POLITE WIFI BEHAVIOR

In this section, we describe the Polite WiFi behavior in more detail and present our test results and findings. We also explain why this problem happens and why it is not preventable.

To better understand the Polite WiFi behavior, we run an experiment where we use two WiFi devices to act as a victim and an attacker. For the victim, we use a tablet, and for the attacker, we use a USB WiFi dongle that has a Realtek RTL8812AU 802.11ac chipset. This is a $12 commodity WiFi device. The attacker uses this device to send fake frames to the victim device. To do so, we develop a simple python program that uses the Scapy library [20] to creates fake frames. Scapy is a python-based framework that can generate arbitrary frames with custom data in the header fields. Note, the only valid information in the frame is the destination MAC address (i.e., the victim's MAC address). The transmitter MAC address is set to the fake MAC address (aa:bb:bb:bb:bb:bb), and the frame has no payload (i.e., null frame) and is not encrypted.

Figure 2 shows the real traffic between the attacker and the victim device captured using Wireshark packet sniffer [9]. As can be seen, when the attacker sends a fake frame to the victim, the victim sends back an ACK to the fake MAC address (aa:bb:bb:bb:bb:bb). This experiment confirms that WiFi devices acknowledge any frame without checking its validity. However, to see if the Polite WiFi behavior exists on other WiFi devices, we have repeated this test with a variety of devices with different WiFi chipsets. Some of these devices are listed in Table 1. We pick a variety of devices such as laptops, smart thermostats, tablets, smartphones, and access points. These devices utilize WiFi chipsets from different vendors. Note, target devices are connected to a private network and the attacker does not have their secret key. After performing the same experiment as before, we found that all of these devices show the Polite WiFi behavior. In Section 3, we conduct a large-scale test that includes over 5,000 WiFi devices.

Device	WiFi module	Standard
MSI GE62 laptop	Intel AC 3160	11ac
Ecobee3 thermostat	Atheros	11n
Surface Pro 2017	Marvel 88W8897	11ac
Samsung Galaxy S8	Murata KM5D18098	11ac
Google Wifi AP	Qualcomm IPQ 4019	11ac

Table 1: List of tested chipsets/devices

2.1 Why does Polite WiFi happen?

In wireless networks, the physical layer is responsible for transmitting and receiving WiFi frames over a wireless channel. When the physical layer receives a frame, it checks the correctness of the frame using error checking mechanisms and transmits an ACK if the frame has no error. However, checking the validity of the content of a frame is performed by the MAC and higher layers. We hypothesize that the separation of responsibilities and the fact that the physical layer does not coordinate with higher layers about sending ACKs seem to be the root cause of the Polite WiFi behavior. In the following, we explain an observation that verifies this hypothesis.

We have observed that when some access points receive fake frames, they start sending *deauthentication frames* to the attacker, requesting it to leave the network. Note this makes no sense since the attacking device has never been part of the network and is not authenticated. However, it seems that some access points detect the attacker as a "malfunctioning" device and that is why they send deauthentication frames. Surprisingly, although these access points have detected that they are receiving fake frames from a "malfunctioning" device, we found that they still acknowledge the fake frames!

A sample traffic that demonstrates this behavior is shown in Figure 3. As can be seen, although the access point has already sent three deauthentication frames to the attacker, it still acknowledges the attacker's fake frame. This verifies that sending ACK frames happens automatically in the physical layer without any communication with higher layers. Therefore, the software running on the access points does not prevent physical layer from sending ACKs to fake frames. To take this experiment to the next level, we manually blocked the attacker's fake MAC address on the access point. Surprisingly, we observed that the AP still acknowledges the fake frames. This experiment destroyed the last hope of preventing this attack.

2.2 Why is Polite WiFi not preventable?

In order to prevent this behavior, WiFi devices must verify if the frame is legitimate before sending an ACK. Unfortunately, this is not possible due to the WiFi standard timing requirements. Specifically,

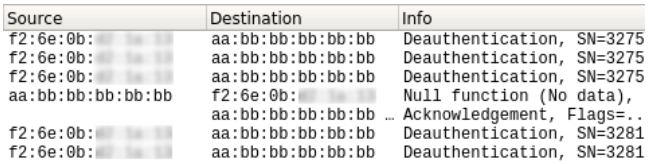


Figure 3: The attacked access point detects that something strange is happening, however it still ACKs fake frames

in the IEEE 802.11 standard, upon receiving a frame, an ACK must be transmitted by the end of the Short Interframe Space (SIFS)[1] interval which is 10 μs and 16 μs for the 2.4 GHz and 5 GHz bands, respectively. If the transmitter does not receive an ACK by the end of SIFS, it assumes that the frame has been lost and retransmits the frame. Therefore, in order to avoid Polite WiFi, WiFi devices need to verify the validity of the received frame in less than 10 μs. This verification must be done by decoding the frame using the secret shared key. Unfortunately, decoding a frame in such a short period is not possible. Prior work has shown that the time required to decode a frame is between 200 to 700 μs when the WPA2 security protocol is used [15, 17, 22]. This processing time is orders of magnitude longer than SIFS. This is exactly why existing devices cannot verify the validity of frame before sending the ACK, and they acknowledge a frame as long as it passes the error detection check.

One may argue that this problem can be addressed by designing a faster security decoder. Unfortunately, even with a faster security decoder, Polite WiFi is still unpreventable since the attacker can send fake Request to Send (RTS) frames instead of fake data frames. Wang et. al [27] show that if a Request to Send (RTS) frame is sent to an unassociated device, it responds with a Clear To Send (CTS) frame. Therefore, if an attacker sends fake RTS frames, the victim responds with CTS frames. The RTS and CTS frames are typically used between WiFi devices in a network to reserve the wireless channel for a certain amount of time. What makes RTS/CTS interesting is that these frames cannot be encrypted in WiFi networks. This is because all nearby devices must receive them to respect channel reservation. The lack of encryption for RTS and CTS frames makes the Polite WiFi behavior unpreventable.[2] For the rest of this paper, we continue using fake frame and ACK for simplicity, although CTS/RTS can be used interchangeably.

3 LARGE-SCALE TESTING

In the previous section, we examined a few different WiFi devices and showed that they are all vulnerable to the Polite WiFi behavior. Here, we examine thousands of devices. In the following, we explain the setup and results of this experiment.

Setup: To examine thousands of devices, we mounted a WiFi dongle on the roof of a vehicle and drove around the city to test all nearby devices. For the WiFi dongle, we use the same Realtek RTL8812AU USB WiFi dongle, and connect it to a Microsoft Surface, running Ubuntu 18.04. We develop a multi-threaded program using the Scapy library [20] to discover nearby devices, send fake 802.11 frames to the discovered devices, and verify that target devices respond to our fake frames. Specifically, our implementation contains three threads. The first thread discovers nearby devices by sniffing WiFi traffic and adding the MAC address of unseen devices to a target list. The second thread sends fake 802.11 frames to the list of target devices. Finally, the third thread checks to verify that target devices respond with an ACK.

WiFi Client Device		WiFi Access Point	
Vendor	# devices	Vendor	# devices
Apple,	143	Hitron	723
Google	102	Sagemcom	601
Intel	66	Technicolor	410
Hitron	65	eero	195
HP	63	Extreme N.	188
Samsung	56	Cisco	156
Espressif	47	HP	104
Hon Hai	46	TP-LINK	101
Amazon	41	Google	80
Sagemcom	38	D-Link	75
Liteon	33	NETGEAR	69
AzureWave	30	ASUSTek	51
Sonos	30	Aruba	46
Nest Labs	27	SmartRG,	44
Murata	24	Ubiquiti N.	35
Belkin	20	Zebra	35
TP-LINK	20	Pegatron	28
Cisco	16	Belkin	25
ecobee	13	Mitsumi	25
Microsoft	13	Apple	19
Others	630	Others	789
Total	1523	Total	3805

Table 2: List of WiFi devices and APs that respond to our fake 802.11 frames.

Results: We perform this experiment for one hour while driving around the city. In total, we discovered 5,328 WiFi nodes from 186 vendors. The list includes 1,523 different WiFi client devices from 147 vendors, and 3,805 access points from 94 vendors. Table 2 shows the top 20 vendors for WiFi devices and WiFi access points in terms of number of devices discovered in our experiment. The list includes devices from major smartphone manufactures (such as Apple, Google, and Samsung) and major IoT vendors (such as Nest, Google, Amazon, and Ecobee). We found that all 5,328 WiFi Access Points and devices responded to our fake 802.11 frames with an acknowledgment, and hence most probably all today's WiFi devices and access points are vulnerable to Polite WiFi.

4 CONSEQUENCES OF POLITE WIFI

In this section, we show why it is important to further study the Polite WiFi behavior. Specifically, we show how an attacker may

[1]The SIFS is used in the 802.11 standard to give the receiver time to go through different procedures before it is ready to send the ACK. These procedures include Physical-layer and MAC-layer header processing, creating the waveform for the ACK, and switching the RF circuit from receiving to transmitting mode.

[2]The IEEE 802.11w standard [1] supports protected management frames to prevent an unauthenticated station from carrying out an attack by injecting fake management frames. However, control frames are still unprotected. Fundamentally, WiFi cannot encrypt control packets because all devices in the vicinity must understand them.

leverage this finding to create privacy and security threats. On the other hand, we also show how a system developer may use this behavior to create new opportunities. Although, we believe our finding opens up many research opportunities, studying all of these opportunities in detail is well beyond the scope of this paper. However, we provide some examples and insights to highlight the importance of Polite WiFi.

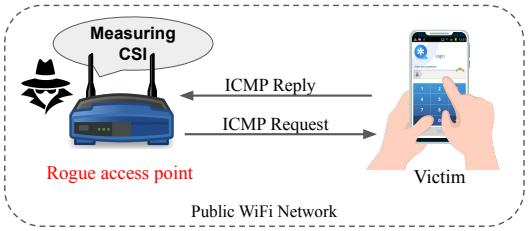
4.1 Privacy Threats

We use keystroke inference as an example to demonstrate new privacy threats created by the Polite WiFi behavior. Recent studies have shown that movements of hand and fingers while typing changes WiFi signals which can be measured by an adversary to reveal what is being typed [4, 12, 16]. This creates a serious threat to the privacy of users because important information such as passwords and private messages could be leaked by monitoring and analyzing the changes of WiFi signals. However, existing keystroke inference attacks (such as WindTalker [16]) either require setting up a dedicated WiFi transmitter and receiver near the target device or need to lure the victim to connect to a rogue access point (AP) as depicted in Figure 4a. In these attacks, when the victim connects to the attacker's AP, the AP starts sending ICMP requests to the victim's device. Then, the device responds to these requests with ICMP reply. Finally, the attacker measures the Channel State Information (CSI) of received ICMP replies to recognize the keystrokes. Although these attacks create serious threats to the privacy of users, convincing the victim to use the attacker's AP is a weak point of theses attacks.

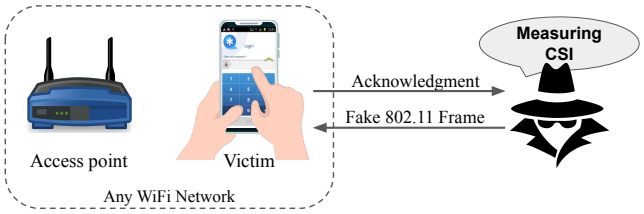
We now explain how an attacker can utilize Polite WiFi to make keystroke inference attack easier, without requiring the victim to connect to the attacker's AP. As shown earlier, we have found that all WiFi devices transmit an acknowledgement for any packet they receive. Hence, an attacker can utilize this finding to eliminate the need for a rogue access point. In particular, as illustrated in Figure 4b, an attacker can send fake 802.11 frames to a victim device and measures the CSI of received acknowledgements. Note, this process substitutes exchanging ICMP packets in past work (such as WindTaker) which requires the victim device to connect to the attacker's AP. Therefore, in contrast to past work, Polite WiFi makes keystroke inference attacks much easier in two ways. First, the attacker does not need the secret key of the victim's WiFi network to perform this attack. Second, the attacked does not need to convince the victim to connect to any particular access point. In fact, even if the victim device is not connected to any WiFi network, this attack still works.

To evaluate the feasibility of this attack, we conduct an experiment in which a target device is connected to an access point. We use a Microsoft Surface Pro tablet as the target device. We then place an attacker device in a different room. We have implemented the attacking mechanism on an ESP32 WiFi module [10] which costs a few dollars[3]. The attacker has no access to the victim's network nor it has the secret key of this network. The attacker sends 150 fake 802.11 frames per second (i.e., null frames with no

[3]We use ESP32 instead of the more commonly used CSI tool [14] and Intel 5300 WiFi card because it enables us to measure the CSI for legacy 802.11a/g bitrates. This is an important feature for us because ACKs are transmitted using legacy bitrates which do not work with the CSI tool.



(a) Existing keystroke inference attacks



(b) Keystroke inference using Polite WiFi

Figure 4: Polite WiFi makes some of existing security attacks extremely more dangerous since the attacker does not need to lure the victim into connecting to a rogue AP.

encryption) to the target device, and measures the CSI of received ACK frames.

Figure 5 shows the CSI amplitude of the signal received from the target device for subcarrier 17. Most other subcarriers had similar patterns. As can be seen, when the tablet is on the ground, the signal amplitude is very stable. However, as soon as a user approaches the device and picks it up the CSI amplitude experiences large fluctuations. Next, the user holds the tablet for about 10 seconds and then starts typing for about 10 seconds. It is very clear that the patterns of just holding the tablet and typing are very distinct. We repeated this experiment multiple times and we observed similar patterns. Although, analyzing what information can be extracted robustly from these measurements is beyond the scope of this paper, this experiment shows that one can potentially reveal what has been typed on a tablet or a phone by using Polite WiFi.

Here, we have used keystroke inference as a example to demonstrate potential new threats caused by the Polite WiFi behavior. However, the scope of these threats extends to many other applications. For instance, an attacker can send fake frames to the devices inside a building to infer various information about what is happening inside that building by measuring changes in the CSI of responses it gets. We believe that there are many open questions that require future study. For example, can an attacker detect occupancy? Can an attacker detect the activity of people inside a building? Or can an attacker estimate vital signs such as heart rate and breathing rate of people from the CSI of their WiFi devices?

4.2 Battery draining Attack

The next threat caused by Polite WiFi is battery draining attack on battery-operated WiFi devices. The goal of this attack is to drain the battery of a WiFi device by forcing the device to transmit WiFi frames continuously. WiFi transmission typically requires

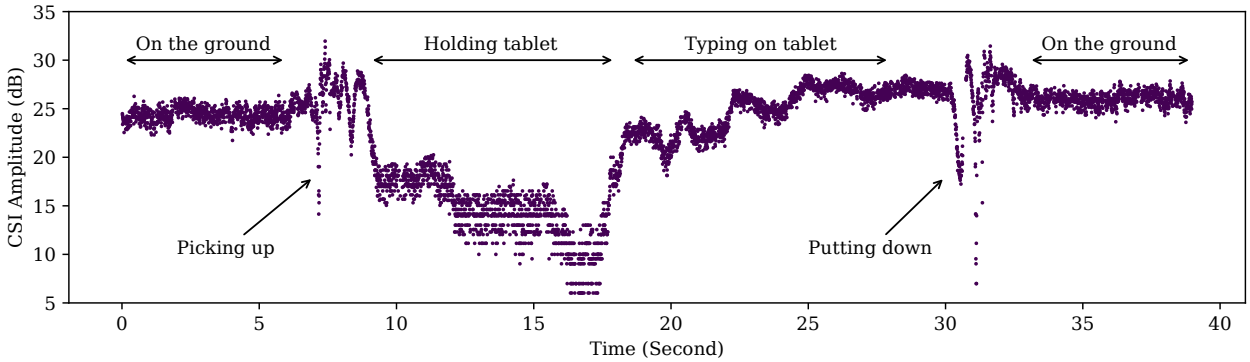


Figure 5: The measured CSI of acknowledgments received from a victim device. The variation of CSI can potentially reveal information such as activities and even the text typed on this device.

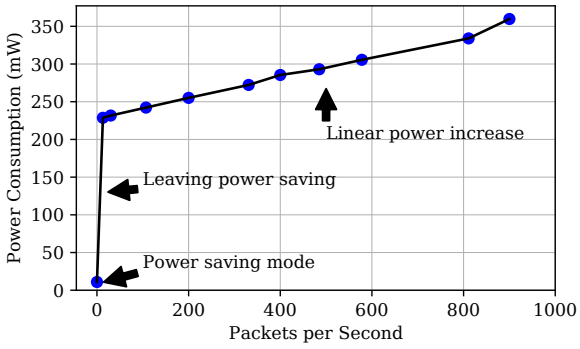


Figure 6: Sending fake frames to a WiFi device increases its power consumption significantly.

several hundreds of milliwatts which puts a huge strain on the battery. To execute this attack, an attacker sends back to back fake 802.11 frames to the target device. This forces the target devices to continuously transmit acknowledgment packets, draining its battery.

To evaluate the efficacy of this attack, we design an experiment where we measure the power consumption of a WiFi module (i.e., a target device) under this attack. We use a Realtek RTL8812AU 802.11ac WiFi USB dongle to inject fake frames, and a Espressif ESP8266 [11] WiFi module as our target device. This module is a low-cost and low-power WiFi microchip used in a variety of IoT devices. In our large-scale testing (i.e., Table 2), we found 47 IoT devices that utilize Espressif WiFi chipsets. We program the module to connect to an access point. The module enters the power saving mode when possible. The power saving mode in WiFi is widely used in battery-operated WiFi devices to extend their battery life since it enables the device to turn off its radio intermittently. In particular, the radio is only powered on for sending a frame or for receiving beacons from the WiFi access point.

Figure 6 shows the power consumption of the target device. If the attacker does not transmit any frame to the target device, the target is mostly in the sleep mode, and hence it consumes only 10 mW. However we found that when the attacker sends more than 10 packets per second to the target device, it prevents the device from going to the sleep mode. Therefore the radio stays on all the time. As the figure shows, this significantly increases the power consummation of the target device to about 230 mW. Note, when the attacker increases the rate of sending fake packets to the target device, the target device has to receive more packets and consequently transmit more ACKs which results in higher power consumption. As shown in Figure 6, the power consumption increases linearly with the rate of packets. For example, when the attacker sends 900 fake packets per second, the power consumption of the target device increases to 360 mW which is a 35x increase in the power consumption compared to when no attack is carried out.

To put these power consumption numbers into perspective, we study the impact of this attack on the battery life of some IoT devices. For instance, the Logitech Circle 2 [19] and Amazon Blink XT2 [7] wireless security cameras have WiFi modules and run on 2400 mWh and 6000 mWh batteries. The Circle 2 camera is advertised to run up to 3 months on a battery, and XT2 is claimed to run up to 2 years. Our measurements on the ESP8266 WiFi module shows that if the attacker transmits 900 frames per seconds to the target device, the target's power consumption increases to 360 mW. As a result, if a similar attack is carried out on these devices, the battery of the Logitech Circle 2 and Blink XT2 security cameras are expected to drain in about 6.7 and 16.7 hours[4]. Draining the battery of these cameras in a short period of time (compared with their expected battery life) is an important security issue. A detailed study of the impact of this attack on the battery life of different IoT and medical devices is an interesting topic for future research.

4.3 New Opportunities for WiFi Sensing

WiFi sensing (such as gesture recognition, occupancy detection, etc.) has recently received significant interest from the research community. Existing WiFi sensing systems typically require two devices to operate, one for transmitting WiFi packets and another one for receiving WiFi signals [30]. By analyzing the change in the received signal, these systems can sense different movements in the environment. However, for these systems to effectively work,

[4]The WiFi module in these security cameras are slightly different from our ESP8266 module. However, both modules are low-power WiFi modules suitable for IoT applications.

they require the target person to be approximately in the line-of-sight between the two WiFi devices [26]. Therefore, in order to perform WiFi sensing for a large area such as an entire house, these systems require multiple devices to cover the whole area. Although, there is already many WiFi devices in current houses, using all these devices for sensing is not feasible. This is due to the fact that WiFi sensing techniques require typically 100 to 1000 packets per seconds to operate [13, 24, 25]. This rate is much more than natural traffic for many WiFi devices. Hence,existing WiFi sensing systems require modification to WiFi devices to force them to transmit packet frequently. Unfortunately, changing the software on access points and some devices like IoT sensors and smart TVs might be difficult or even impossible.

Polite WiFi solves this challenge by enabling WiFi sensing with software modification on only one device. For example, one device such as an IoT hub, can transmit fake 802.11 frames to nearby WiFi devices and measure the CSI of the received ACKs. Although, other devices are participating in these measurements, there is no software modification required on them. Therefore, any nearby WiFi device can be utilized in WiFi sensing applications. Figure 5 shows simple example of WiFi sensing using Polite WiFi where movements near the target WiFi device has created sharp changes in the CSI amplitude at times 9 and 32. More detailed information can be potentially inferred using techniques proposed in recent WiFi sensing studies. Studying these possibilities is an interesting topic for future research.

5 RELATED WORK

To the best of our knowledge, the Polite WiFi behavior has not been studied before and therefore there is no study directly related to this work. In the following we review studies that use techniques similar to Polite WiFi in different contexts. Wang et al [27] has previously used the RTS/CTS exchange. However, their focus is for detecting unassociated WiFi devices in the context of aerial search and rescue operations. There is also past work that utilize frame injection to create new opportunities and threats for WiFi networks. For instance, frame injection has been utilized to enable new features such as beacon stuffing and low-power WiFi communication. In beacon stuffing, 802.11 beacon frames, that normally carry access point information, are modified to carry some other useful information [8, 29]. These fake beacon frames enable application such as location-specific advertisements and providing coupons over WiFi without the need for association. Frame injection has also been utilized to enable low-power WiFi communication that lowers the power consumption of WiFi to that of Bluetooth [2]. In this technique, the overhead of establishing and maintaining a WiFi connection is avoided by injecting broadcast frames without associating with an AP.

Packet injection has also been used to perform various types of attacks against WiFi networks, such as Denial of Service (DoS) attack [3], rogue access points [5], and deauthentication attack [6]. However, all of these attacks focus on spoofing 802.11 MAC-layer management frames to interrupt the normal operation of WiFi networks. For example, in the deauthentication attack, the victim device is dropped from its network by injecting fake deauthentication frames to the access point. To provide a countermeasure for some of these attacks, the 802.11w standard [1] introduces protected management frame which prevents attackers from spoofing 802.11 management frames.

In contrast, Polite WiFi is fundamentally different from prior work. Instead of spoofing 802.11 MAC frames, we exploit properties of the 802.11 physical layer to force a device to send an acknowledgment. The Polite WiFi behavior opens the door to multiple research avenues including new security and privacy threats and new opportunities for WiFi sensing applications.

6 CONCLUSION

We have discovered and explored Polite WiFi, a behavior in which WiFi devices transmit an ACK for any 802.11 frame they receive. Our evaluation of over 5,000 devices from 186 vendors confirms that this is a widespread issue. Moreover, it is not preventable because of timing restrictions in the physical layer. We believe our findings open up many research directions to study the possible threats and opportunities created by the Polite WiFi behavior. To show the importance of this behavior and its consequences, in this paper, we show how Polite WiFi can be coupled with WiFi sensing techniques to steal private personal information such as passwords. We also demonstrate how Polite WiFi can be exploited to increase the power consumption of a WiFi chipset by 35 times to drain the battery of a WiFi device quickly. Finally, we also describe how Polite WiFi creates new opportunities for WiFi sensing techniques by making their deployment more practical.

ACKNOWLEDGMENTS

We thank the anonymous reviewers for their helpful feedback.

REFERENCES

[1] IEEE Standard for Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements. Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 4: Protected Management Frames. *IEEE Std 802.11w-2009 (Amendment to IEEE Std 802.11-2007 as amended by IEEE Std 802.11k-2008, IEEE Std 802.11r-2008, and IEEE Std 802.11y-2008)*, pages 1–111, 2009.

[2] A. Abedi, O. Abari, and T. Brecht. Wi-LE: Can WiFi Replace Bluetooth? In *Proceedings of the 18th ACM Workshop on Hot Topics in Networks*, HotNets '19, page 117–124, 2019.

[3] M. Agarwal, D. Pasumarthi, S. Biswas, and S. Nandi. Machine learning approach for detection of flooding DoS attacks in 802.11 networks and attacker localization. *International Journal of Machine Learning and Cybernetics*, 7:1035–1051, 2016.

[4] K. Ali, A. X. Liu, W. Wang, and M. Shahzad. Keystroke Recognition Using WiFi Signals. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*, MobiCom '15, page 90–102, 2015.

[5] B. Alotaibi and K. Elleithy. Rogue Access Point Detection: Taxonomy, Challenges, and Future Directions. *Wireless Personal Communications*, 90:5021– 5028, 10 2016.

[6] J. Bellardo and S. Savage. 802.11 Denial-of-Service Attacks: Real Vulnerabilities and Practical Solutions. *Proceedings of 12 USENIX Security Symposium*, 2003.

[7] Blink. *Blink XT2 security camera*, 2020. https://blinkforhome.com/products.

[8] R. Chandra, J. Padhye, L. Ravindranath, and A. Wolman. Beacon-stuffing: Wi-fi without associations. In *Eighth IEEE Workshop on Mobile Computing Systems and Applications*, pages 53–57, 2007.

[9] G. Combs. *Wireshark*, 2020. https://www.wireshark.org/.

[10] Espressif Systems. *ESP32 datasheet*, 4 2019. https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf.

[11] Espressif Systems. *ESP8266 datasheet*, 4 2020. https://www.espressif.com/sites/default/files/documentation/0a-esp8266ex_datasheet_en.pdf.

[12] S. Fang, I. Markwood, Y. Liu, S. Zhao, Z. Lu, and H. Zhu. No Training Hurdles: Fast Training-Agnostic Attacks to Infer Your Typing. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, CCS '18, page 1747–1760, 2018.

[13] X. Guo, B. Liu, C. Shi, H. Liu, Y. Chen, and M. C. Chuah. WiFi-Enabled Smart Human Dynamics Monitoring. In *Proceedings of the 15th ACM Conference on Embedded Network Sensor Systems*, SenSys '17, 2017.

[14] D. Halperin, W. Hu, A. Sheth, and D. Wetherall. Tool release: Gathering 802.11n traces with channel state information. *ACM SIGCOMM CCR*, 41(1):53, Jan. 2011.

[15] S. S. Kolahi and A. A. Almatrook. Impact of security on bandwidth and latency in IEEE 802.11ac client-to-server WLAN. In *2017 Ninth International Conference on Ubiquitous and Future Networks (ICUFN)*, pages 893–897, 2017.

[16] M. Li, Y. Meng, J. Liu, H. Zhu, X. Liang, Y. Liu, and N. Ruan. When CSI Meets Public WiFi: Inferring Your Mobile Phone Password via WiFi Signals. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, CCS '16, page 1068–1079, 2016.

[17] P. Li, S. S. Kolahi, M. Safdari, and M. Argawe. Effect of WPA2 Security on IEEE 802.11n Bandwidth and Round Trip Time in Peer-Peer Wireless Local Area Networks. In *2011 IEEE Workshops of International Conference on Advanced Information Networking and Applications*, pages 777–782, 2011.

[18] J. Liu, Y. Wang, Y. Chen, J. Yang, X. Chen, and J. Cheng. Tracking Vital Signs During Sleep Leveraging Off-the-Shelf WiFi. In *Proceedings of the 16th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, MobiHoc '15, page 267–276, 2015.

[19] Logitech. *Circle 2 security camera*, 2020. https://www.logitech.com/en-ca/product/circle-2-home-security-camera.

[20] Philippe Biondi. *Scapy*, 2020. https://scapy.net/.

[21] A. Rai, K. K. Chintalapudi, V. N. Padmanabhan, and R. Sen. Zee: Zero-Effort Crowdsourcing for Indoor Localization. In *Proceedings of the 18th Annual International Conference on Mobile Computing and Networking*, Mobicom '12, page 293–304, 2012.

[22] M. Saleh, J. Gaber, and M. Wack. Sensor Networks Applications Performance Measures for IEEE802.11n WiFi Security Protocols. In *Proceedings of the International Conference on Future Networks and Distributed Systems*, ICFNDS '17, 2017.

[23] D. Vasisht, S. Kumar, and D. Katabi. Decimeter-Level Localization with a Single WiFi Access Point. In *Proceedings of the 13th Usenix Conference on Networked Systems Design and Implementation*, NSDI'16, page 165–178, 2016.

[24] R. H. Venkatnarayan, G. Page, and M. Shahzad. Multi-User Gesture Recognition Using WiFi. In *Proceedings of the 16th Annual International Conference on Mobile Systems, Applications, and Services*, MobiSys '18, page 401–413, 2018.

[25] A. Virmani and M. Shahzad. Position and Orientation Agnostic Gesture Recognition Using WiFi. In *Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services*, MobiSys '17, page 252–264, 2017.

[26] H. Wang, D. Zhang, J. Ma, Y. Wang, Y. Wang, D. Wu, T. Gu, and B. Xie. Human Respiration Detection with Commodity Wifi Devices: Do User Location and Body Orientation Matter? In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '16, page 25–36, 2016.

[27] W. Wang, R. Joshi, A. Kulkarni, W. K. Leong, and B. Leong. Feasibility study of mobile phone wifi detection in aerial search and rescue operations. In *Proceedings of the 4th Asia-Pacific Workshop on Systems*, APSys '13, 2013.

[28] S. Yun, Y.-C. Chen, and L. Qiu. Turning a Mobile Device into a Mouse in the Air. In *Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services*, MobiSys '15, page 15–29, 2015.

[29] S. Zehl, N. Karowski, A. Zubow, and A. Wolisz. Lows: A complete open source solution for wi-fi beacon stuffing based location-based services. In *2016 9th IFIP Wireless and Mobile Networking Conference (WMNC)*, pages 25–32, 2016.

[30] Y. Zheng, Y. Zhang, K. Qian, G. Zhang, Y. Liu, C. Wu, and Z. Yang. Zero-Effort Cross-Domain Gesture Recognition with Wi-Fi. In *Proceedings of the 17th Annual International Conference on Mobile Systems, Applications, and Services*, MobiSys '19, 2019.