MakeShift: Security Analysis of Shimano Di2 Wireless Gear Shifting in Bicycles

Maryam Motallebighomi, *Northeastern University;* Earlence Fernandes, *UC San Diego;* Aanjhan Ranganathan, *Northeastern University*

https://www.usenix.org/conference/woot24/presentation/motallebighomi

This paper is included in the Proceedings of the 18th USENIX WOOT Conference on Offensive Technologies.

August 12–13, 2024 • Philadelphia, PA, USA

ISBN 978-1-939133-43-4

Open access to the Proceedings of the 18th USENIX WOOT Conference on Offensive Technologies is sponsored by USENIX.

MakeShift: Security Analysis of Shimano Di2 Wireless Gear Shifting in Bicycles

Maryam Motallebighomi[1], Earlence Fernandes[2], and Aanjhan Ranganathan[1]

[1]Northeastern University, Boston, USA
[2]University of California, San Diego, USA

Abstract

The bicycle industry is increasingly adopting wireless gear-shifting technology for its advantages in performance and design. In this paper, we explore the security of these systems, focusing on Shimano's Di2 technology, a market leader in the space. Through a blackbox analysis of Shimano's proprietary wireless protocol, we uncovered the following critical vulnerabilities: (1) A lack of mechanisms to prevent replay attacks that allows an attacker to capture and retransmit gear shifting commands; (2) Susceptibility to targeted jamming, that allows an attacker to disable shifting on a specific target bike; and (3) Information leakage resulting from the use of ANT+ communication, that allows an attacker to inspect telemetry from a target bike. Exploiting these, we conduct successful record and replay attacks that lead to unintended gear shifting that can be completely controlled by an attacker without the need for any cryptographic keys. Our experimental results show that we can perform replay attacks from up to 10 meters using software-defined radios without any amplifiers. The recorded packets can be used at any future time as long as the bike components remain paired. We also demonstrate the feasibility of targeted jamming attacks that disable gear shifting for a specific bike, meaning they are finely tuned to not affect neighboring systems. Finally, we propose countermeasures and discuss their broader implications with the goal of improving wireless communication security in cycling equipment.

1 Introduction

Modern bicycles are cyber-physical systems that contain embedded computers and wireless links to enable new types of telemetry and control. The key motivating factors for moving away from traditional mechanical systems are the ability to gain insights about a rider's physical performance, better responsiveness in gear shifting, customizability of how the gear shifters operate, and easier setup and maintenance.

Among all these technologies, we observe that the one with the most impact on bike control and safety is wireless gear shifting.[1] It uses wireless links between the gear shifters and the derailleur — an electro-mechanical component that uses motors to move the chain between gears. Electronic control provides increased precision in shifts and is less prone to issues like cable stretch and contamination that plague mechanical gear shifting systems. Although wired electronic control of gear shifting exists, the current trend in the bicycle industry is to move towards wireless control. All major manufacturers now support wireless shifting (Shimano, SRAM, Campagnolo).

In this work, we analyze the security guarantees of wireless gear shifting. Any security vulnerability in this system can significantly impact the rider's safety and performance, especially in professional bike races, where an attacker could target a victim rider to gain an unfair competitive advantage. In a professional race, a peloton of hundreds of riders are close to each other, often a few feet apart, and can reach speeds up to 40 mph. Any sudden changes to a bike's performance can be catastrophic. For example, if an attacker were to target a subset of riders and shift the gears or jam the shifting operation, it could result in crashes and injuries. As another example, if the riders are climbing slowly (or descending at high speed), an attacker could shift a target rider's bike into high gear or jam their shifting, leading them to lose their position in the race and even lose control of the bike itself.

The sport of professional cycling has a long and troubled history with the use of illegal performance-enhancing drugs — security vulnerabilities in one of the most critical components of the bike could be viewed as an attractive alternative method for people who want to compromise the integrity of the sport. Furthermore, our attacks do not leave any detectable trace, unlike the use of performance-enhancing drugs. As such, with the introduction of wireless gear shifting, one must adopt an adversary's viewpoint — professional bike races are adversarial environments, and the technology must withstand motivated attackers. We focus on the Shimano 105

[1]The other important component is the brakes, but these are mechanical systems.

Di2 [10] and Shimano DURA-ACE Di2 [16] wireless shifting systems. Shimano is a leader in the bicycle control system industry, commanding approximately 50% of the market share [15, 28]. We purchased a recent version of the control system and performed a black box security analysis, from capturing raw physical signals, examining their behavior on gear shifting, and finally performing packet structure/content analysis. This study seeks to address the following research questions: (1) What are the security guarantees provided by these wireless gear-shifting systems? (2) Do these wireless systems, when integrated into bicycles, maintain robust defenses against specific cyber attacks, such as replay attacks, similar to those observed in automotive key fob systems [20]? Have the lessons learned from analyzing similar systems contributed to the design of these wireless gear shifters? (3) What is the practical feasibility of executing the identified cyber attacks? In other words, what constraints and requirements would an attacker face in attempting to compromise these systems?

Our key contribution is the discovery of a record-and-replay attack that allows an unauthorized party to fully control gear shifting on a victim bike at ranges up to 10 meters without the use of amplifiers. This attack can be realized using commercial-off-the-shelf software-defined radios (SDR). The attacker only needs to record two signals — an upshift and a downshift.

We make the following contributions in this paper.

- **Analysis of the Shimano Wireless Gear Shifting Protocol.** We investigate the proprietary protocol used by Shimano for its wireless gear shifting. This process allows us to decode the communication framework of these systems, providing insights into their operational mechanics.
- **Identify Security Weaknesses.** Based on the analysis, we identify several security weaknesses within the protocol, notably the absence of replay protection mechanisms such as timestamps or sequence numbers. Despite the implementation of cryptographic primitives, these vulnerabilities present significant security risks.
- **Record-and-Replay and Targeted Jamming Attacks.** Leveraging the identified weaknesses, we successfully execute record-and-replay attacks. These attacks can cause unexpected gear shifts in arbitrary patterns by interacting at the physical layer, bypassing the need for extracting any cryptographic secrets and making the attack independent of the cryptographic layer. Furthermore, we explore the potential for targeted jamming attacks that specifically disable gear-shifting capabilities on targeted bicycles without impacting nearby cyclists.
- **Experimental Evaluation.** We conduct various experiments with two identical Shimano 105 Di2 wireless gear shifting systems. We also confirmed our findings on Shimano DURA-ACE Di2 system. Through these experiments, we executed replay and jamming attacks utilizing SDRs and explored their effective range. Additionally, we examined the shifting system's behavior in response to interference. Our experiments indicate that replay attacks using SDRs are effective up to a distance of 10 meters without amplification. The effectiveness of replayed packets persists as long as the pairing between shifters and derailleur remains unchanged.
- **Countermeasures.** We provide a discussion of potential countermeasures. Wireless gear shifting operates in a highly constrained environment — security mechanisms should not add significant time delay in shifting and must not degrade battery life. While implementing techniques such as timestamps has particular challenges, employing rolling codes or distance bounding within wireless gear shifting can effectively mitigate replay attacks.

Although our paper's main focus is on Shimano's gear-shifting systems, we also examine vulnerabilities in the communication protocol used for telemetry on bike displays, notably the ANT protocol. This protocol is widely used in Shimano and other low-power wireless data transmission systems, extending the relevance of our findings. We have shown that any nearby third party with Shimano's private key and knowledge of the channel configuration can intercept all transmitted information. In our replay attack scenario, this information enables the attacker to determine the targeted bike's current gear and replay the upshifting/downshifting commands to adjust the gear according to their preference. For example, the attacker waits until the rider is in gear 3 and then launches a downshift replay to move it to gear 2.

Our study aims to highlight vulnerabilities in wireless gear shifting systems, especially focusing on Shimano's Di2, and offers a first look into the security challenges of bicycle wireless communication technologies. Through this work, we hope to contribute to the ongoing effort to secure wireless communications in cycling equipment.

Responsible Disclosure. We notified Shimano about the vulnerabilities, along with detailed information on replicating the attacks, part numbers of the devices we tested, and a description of countermeasures that might be helpful in this context. Shimano has acknowledged these vulnerabilities and is working on fixes at the time of this writing.

2 Wireless Gear Shifting: An Overview

In the cycling industry, all major manufacturers have ventured into developing wireless gear-shifting systems, aiming to enhance the cycling experience through technology. Brands like SRAM [17] and Campagnolo [3], alongside Shimano, have introduced their versions of wireless shifting, each bringing unique features and innovations to the market. These systems signify a leap forward in bicycle design, offering cyclists

Table 1: The equipment list on the test bikes for Shimano 105 Di2 groupset.

Item	Model	Firmware Version
Rear Derailleur	RD-R7150	ver 4.0.2
Front Derailleur	FD-R7150	ver 4.0.1
Right Shifter	ST-R7170-R	-
Left Shifter	ST-R7170-L	-
Battery	BT-DN300	ver 4.0.1

Table 2: The equipment list on the test bikes for Shimano Dura-Ace Di2 groupset.

Item	Model	Firmware Version
Rear Derailleur	RD-R9250	ver 4.0.7
Front Derailleur	FD-R9250	ver 4.0.3
Right Shifter	ST-R9270-R	-
Left Shifter	ST-R9270-L	-
Battery	BT-DN300	ver 4.0.1

improved performance, convenience, and integration. Due to Shimano's significant market presence and role as a pioneering force in cycling technology, we've selected Shimano as our case study to examine the vulnerabilities inherent in wireless gear-shifting systems.

For our experiments, we chose the Shimano 105 Di2 and the Dura-Ace Di2 wireless gear-shifting systems as our case study. Tables 1 and 2 contain the equipment list, their respective model numbers, and firmware versions. We note that the two groupsets, Shimano 105 Di2 and Shimano Dura-Ace Di2, are compatible. Therefore, we tested pairing various shifters and derailleurs from these groupsets and confirmed that they all use the same protocol. Consequently, the vulnerabilities identified are consistent across both systems.

The Shimano gear-shifting system consists of four main components.

(1) Rear Derailleur: The rear derailleur is the core of the gear-shifting system and facilitates all wireless communications. This includes connections with the shifters, Bluetooth Low Energy (BLE), and ANT+ communications. It offers eleven gear levels (and, in some newer versions, 12 levels), ranging from the lowest to the highest.

(2) Front Derailleur: The front derailleur is wired to the rear derailleur through the battery and allows switching between two distinct gear levels, which are large gear changes.

(3) Right and Left Shifters: These components wirelessly transmit gear-shifting instructions to the rear derailleur using Shimano's proprietary protocol, which we will explore in detail in the following section. One of the shifters controls the rear derailleur, and the other controls the front derailleur. This setting can be customized through the E-TUBE PROJECT [4] over BLE.

(4) Battery: The battery ports are connected to the rear and front derailleur, ensuring they are powered for operation.

Figure 1: Shimano's RF communication.

The Shimano system employs three key protocols to establish connections among its various components, each serving a distinct function. The communication methods within Shimano's network are illustrated in Figure 1.

2.1 Bluetooth Low Energy

The Shimano E-TUBE PROJECT is a software tool that connects cyclists to their bike configuration. This platform can personalize the settings, such as customizing shifter button functions and conducting firmware updates. It employs BLE for efficient communication in many power-constrained devices, which fits the requirements of a system like E-TUBE PROJECT that aims to provide seamless and user-friendly interaction with bicycle components. While BLE is essential for configuring and updating the system, it *does not* control real-time biking actions such as shifting gears.

Also, the initial setup of shifters and the rear derailleur involves pairing them through the E-TUBE mobile app. Users need to register and connect the rear derailleur to the mobile app, then scan the QR code on the shifters to pair both shifters with the rear derailleur. Given that Bluetooth Low Energy (BLE) vulnerabilities have been extensively documented in existing literature [41], our paper did not focus on this aspect.

2.2 ANT+

ANT is a low-power wireless protocol designed to transmit information between devices efficiently and reliably. It is known for robustness and adaptability in different network setups, including mesh networks, making it ideal for gathering and sending sensor data.

Building on the ANT protocol, ANT+ is an enhancement that standardizes how specific data types are communicated. It establishes device profiles for consistent data transmission, like heart rate, bike speed, and cadence. This standardization allows devices from various manufacturers to work together seamlessly. In cycling, ANT+ plays a crucial role in the Di2 system. It wirelessly sends vital information such as gear position and battery life to compatible cycling computers,

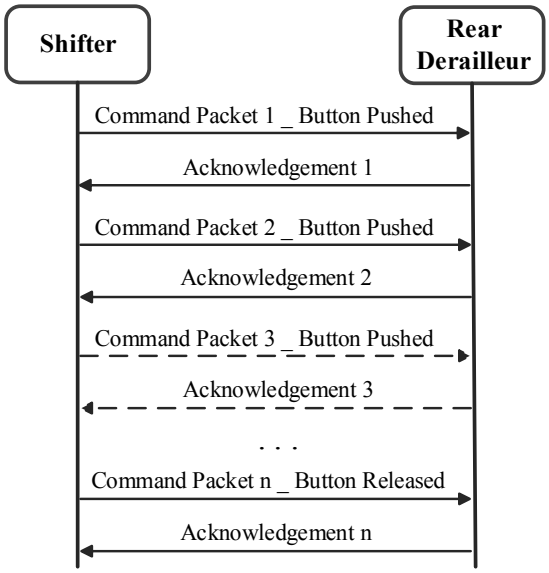

Figure 2: The sequence of command and acknowledgment packets between the shifter and rear derailleur after a button press. The user's actions can influence the sequence and number of command packets, which are subsequently followed by an acknowledgment.

allowing riders to monitor these details in real-time during their rides. The frequency range for ANT devices spans from 2.400 GHz to 2.524 GHz, but 2.457 GHz is reserved specifically for ANT+ devices. These devices can operate using a public network key, a private one, or a managed network key owned privately, providing flexibility in network security and access [1]. In summary, ANT and ANT+ offer versatile and efficient solutions for wireless communication, especially in scenarios where reliable data transmission and interoperability are essential.

2.3 Shimano's Proprietary Communication Protocol

In the Shimano Di2 system, gear shifting is controlled through a unique, Shimano-specific protocol. This protocol operates on the 2.478 GHz frequency band, facilitating communication between the rear derailleur and the shifters. However, Shimano's official documentation does not disclose detailed information about this protocol, leaving specifics such as modulation, data rate, and packet structure unclear. Thus, we analyze Shimano's proprietary communication protocol as a first step.

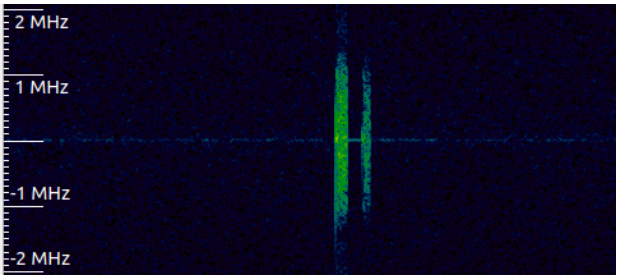

Figure 3: One command packet being transmitted from the shifter to the rear derailleur, along with the corresponding acknowledgment sent from the rear derailleur back to the shifter.

3 Analyzing Shimano's Wireless Gear Control Protocol

We begin with an overview of Shimano's Wireless Gear Control Protocol, providing a detailed examination of the command and acknowledgment packet sequences exchanged between the shifter and the rear derailleur. Next, we analyze the physical layer, focusing primarily on demodulating captured RF signals into binary data to understand the underlying communication mechanisms. We employ a black-box methodology to passively capture raw signals. Subsequently, we delve into the packet structures within the Shimano wireless communication protocol, exploring all components of the various packet types. Finally, we discuss the security weaknesses that could potentially threaten the protocol's integrity.

3.1 High-Level Protocol Overview

The shifters send two types of commands to the rear derailleur — Gear Up and Gear Down. On each press of the shift button (either up or down), the shifter transmits at least three packets to the derailleur. Upon receiving each packet, the derailleur transmits an acknowledgment to the shifter. The quantity of packets transmitted is influenced by the speed at which the user presses and releases the shifter button. If the button is pressed and held, packets will be sent for the hold duration. Conversely, a single press of the button results in the transmission of at least three packets. Figure 2 illustrates the sequence of packets triggered by the user pressing the button, leading to one upshift on the rear derailleur. As noted, the sequence of command packets followed by an acknowledgment can vary based on the user's actions. Figure 3 displays one command packet being transmitted from the shifter to the rear derailleur, along with the corresponding acknowledgment sent from the rear derailleur back to the shifter. Each command packet has an approximate duration of 112 µs, while each acknowledgment packet is about 76 µs.

If the shifter fails to receive an acknowledgment within a time frame, it initiates a burst transmission. Each burst

Table 3: Behavior of different packets during a replay attack: Pressing the button by the user results in the transmission of three packets from the shifter to the derailleur. To understand the packet's functionality, we conducted experiments by replaying the packets both individually and in various combinations.

Setting	First Packet	Second Packet	Third Packet	Observations
A	1			This will cause the derailleur to shift up by one gear. The repeating signal will not function until you manually press the button once. After that, the signal can be replayed successfully once more.
B		1		Similar to A.
C			1	No reaction.
D	1		1	Works like a normal replay. Repeated many times.
E		1	1	Similar to D.
F	1	1		Every time the signal was replayed, it resulted in shifting twice instead of a single time.

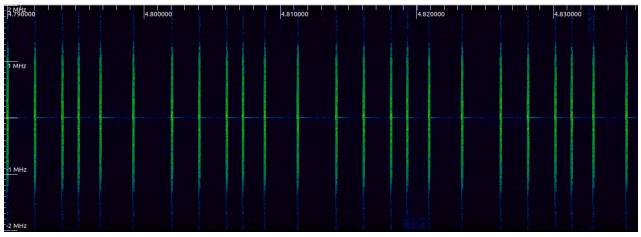

Figure 4: Segments from a burst sequence when acknowledgments are not received, showcasing a total of 748 packets transmitted over 1.5 seconds.

contains 748 packets and lasts 1.5 seconds. We captured the packet burst while the rear derailleur was disconnected from the battery. Figure 4 illustrates a segment of this burst. In Section 4.3, we discuss the relevance of the packet burst under conditions like interference.

In the next step, we conducted tests by individually transmitting the three captured packets associated with a single gear shift to analyze each packet's effect. Furthermore, we experimented with various combinations of these three packets to examine the outcomes, acknowledging that redundancy among them might be designed to guarantee command reception by the derailleur to prevent potential interference.

Table 3 summarizes the functionality of the packets. We monitored how the packets behaved under different conditions (labeled A to F), continuously replaying the specific packet(s) relevant to each condition. This was then contrasted with a baseline scenario, wherein all three packets were replayed in their original sequence, mirroring the authentic command exactly. Our observations indicated that the behaviors of the first and second packets were strikingly similar.

On the other hand, replaying only the third packet triggers no derailleur action, leading us to speculate that this packet might serve as a "button released" command.

In scenarios D and E, eliminating the first or second packet does not affect the behavior of the packets compared to the baseline scenario. This suggests that one of the packets may be sent as a form of redundancy. In both cases, D and E, repeatedly replaying the packets consistently triggers a single gear shift, akin to the baseline scenario. However, replaying the same packet in scenarios A and B does not lead to subsequent gear shifts after the successful initial replay.

Furthermore, in scenario F, sending both the first and second packets causes the gear to shift twice, which could mirror the situation where the user keeps the button pressed.

We clarify that our analysis in Table 3 focuses exclusively on individual shift events rather than MultiShift settings. MultiShift settings in Shimano's wireless gear-shifting system allow multiple gear changes with a single button press, enabling quicker transitions across gears. This distinction is important as our experimental setup and data collection were designed to evaluate single-shifting actions.

In conclusion, our experiments revealed that the roles of the first and second packets might stem from redundancy and correspond to the user's button press, while the third packet appears to be associated with the user releasing the button.

Table 4: Signals information derived from publicly available documents

Signal Feature	Value
Frequency	2.478 GHz
Bandwidth	2127 KHz
Modulation	GFSK
Emission Reference	<TX3064779>
Emission Designator	2M13F1D–

3.2 Physical Layer Analysis

The primary focus is demodulating the captured RF signals into binary data and subsequently examining their contents. We use a black box methodology that passively captures raw signals.

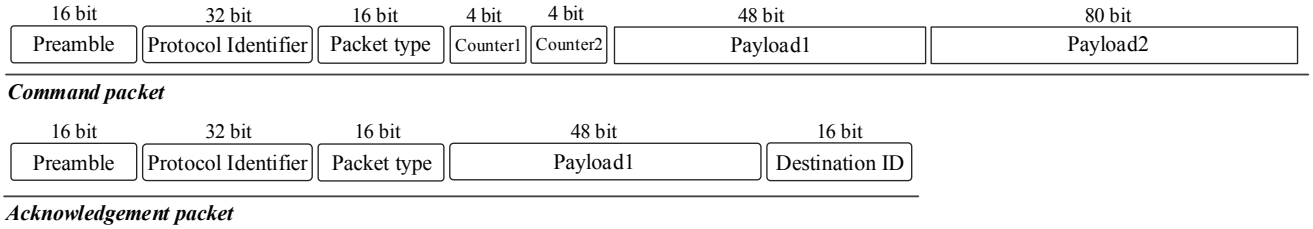

Figure 5: Overview of Command and Acknowledgment Packet Structures. Illustrating the differences between 200-bit command packets and 128-bit acknowledgment packets, including the content of their respective fields.

Table 5: Modulation/Demodulation parameters for Shimano's proprietary protocol

Modulation Parameter	Value
Carrier Frequency	2.478 GHz
Data Rate	2 Mbps
Bit per Symbol	1
Frequency Deviation	-250 kHz/250 kHz
Gauss BT	0.5
Gauss Filter Width	1

The first step in analyzing a wireless signal is determining its precise frequency and modulation type. We found this data in documents from the Federal Communications Commission (FCC) and the Radio Equipment List (REL). Table 4 presents the information summarized from these documents. The communication between devices occurs at 2.478GHz and does not utilize frequency hopping. The signal's bandwidth is 2127 KHz.

The term 'Emission Reference <TX3064779>' is identified as a distinct code or number linked to specific emission properties. The 'Emission Designator 2M13F1D' is recognized globally to categorize a signal's bandwidth, modulation type, and content. '2M13' details the required signal bandwidth. 'F' denotes the modulation type of the primary carrier as frequency modulation, '1' represents a single channel carrying quantized or digital data without an additional modulating sub-carrier, and 'D' describes the nature of the transmitted information, highlighting the transmission of digital data.

Shimano gear shifting utilizes Gaussian Frequency Shift Keying (GFSK) for their proprietary communication protocol, a form of Frequency Shift Keying that applies Gaussian filtering to smooth out signal transitions or frequency shifts. GFSK is a prominent modulation technique employed across various wireless technologies, including Bluetooth, IEEE 802.15.4, and Z-wave. Dealing with GFSK presents more complexity in the analysis compared to systems using simpler modulation techniques like amplitude shift keying, where signal demodulation can be straightforwardly achieved using open-source tools such as Inspectrum [8]. However, demodulating GFSK requires identifying the correct demodulation parameters, which increases the complexity. For Shimano devices, all specific modulation parameters were initially unclear. The FCC documents did not disclose any of these parameters.

We utilized Universal Radio Hacker (URH) [34], a tool specifically designed to analyze and manipulate wireless communication signals for our analysis. This tool facilitates the recording, analysis, and modification of signals across various wireless devices. However, the automatic parameter detection feature in URH failed to demodulate our captured signal effectively. We were unclear about the data rate, a critical piece of information for GFSK demodulation, which depends heavily on the correct sample/symbol ratio. Through a combination of trial and error and visual analysis of our signals, we identified the necessary parameters to successfully demodulate the captured data. Table 5 outlines the required modulation/demodulation parameters we identified. The Gaussian filter in GFSK modulation has a parameter called the time-bandwidth product (BT), which is the product of the filter bandwidth and the bit duration. The BT value affects the shape of the data pulses and the resulting GFSK signal.

3.3 Packet Structure and Content

Upon successful demodulation, we could distinguish two primary types of packets within the Shimano communication protocol: command and acknowledgment. Command packets, originating from the shifter, comprise 200 bits and are directed towards the derailleur. Conversely, acknowledgments follow these command signals and consist of 128 bits, transmitting from the derailleur back to the shifter. Figure 5 graphically illustrates these packet structures, annotated with their components.

In our analysis of numerous packet sequences, we identify and describe specific fields within the packets as follows:
Preamble: Each packet starts with a 16-bit preamble, represented as `0101010101010101`. The preamble plays a critical role in various RF communication protocols by helping to synchronize the receiver's timing with the sender's signal. This synchronization aids in accurately detecting the beginning

Table 6: Analysis of different fields in Shimano command packets. It shows our observation of how different fields in the packets change under various conditions, helping to clarify how the fields are connected.

Action/Condition	Counter1 changes?	Counter2 changes?	Observations		
			Payload1 Consistency	Payload2 Consistency	Destination ID Consistency
Bike1, Upshifting	No	No	Yes	Yes	Yes
Bike1, Upshifting	Yes	No	No	No	Yes
Bike1, Upshifting	No	Yes	No	No	Yes
Bike1, Downshifting	No	No	Yes	No	Yes
Bike2, Upshifting	No	No	No	No	No
Bike1, Upshifting (Repairing)	No	No	Yes	No	Yes

of a new packet. Additionally, the preamble facilitates the adjustment of the receiver's Automatic Gain Control (AGC) circuits to the strength of the incoming signal. We verified the correct demodulation of the preamble at the start of each packet during our adjustments of the modulation parameters.

Protocol Identifier: Following the preamble is a 32-bit protocol identifier, which remains unchanged across all packets captured under this specific Shimano's proprietary protocol. This identifier functions similarly to the "access address" in BLE protocols and helps distinguish Shimano's protocol from other traffic in the 2.4 GHz spectrum.

Packet Type: A 16-bit field follows, identifying the packet as either a command or acknowledgment packet. In our observations, every 200-bit command packet contains `0x8888` within this field, while acknowledgments are marked with `0x1010`.

Counters: The first counter (`Counter1`) increments with each transmitted packet. The second counter (`Counter2`) increments only upon receiving an acknowledgment, indicating a successful transmission. If the shifter does not receive an acknowledgment and begins to emit a burst of packets, `Counter2` remains constant, whereas `Counter1` cycles through 15 possible values.

Payload: The next segment within command packets involves `Payload1` and `Payload2`, together spanning 128 bits. We divided the payload into two parts because some parts of the payload in command packets are exactly replicated in the acknowledgments.

Table 6 offers a comprehensive look at our findings, illustrating how each field varies across different test scenarios and setups. For any given wireless shifting setup and command type (either upshifting or downshifting), `Payload1` and `Payload2` remain consistent as long as the counters are identical.

Specifically, `Payload1` comprises a sequence that is present both in the command packets and in the acknowledgment packets. This section of the payload, a 48-bit sequence, is repeated in the acknowledgment packets to confirm which command the acknowledgment is intended for. On the other hand, `Payload2` is exclusive to the command packets and does not appear in the acknowledgment packets. If two packets have similar values for counter1 and counter2, the values of both `Payload1` and `Payload2` would be the same too. This consistency holds true while the same shifters are consistently paired with the same derailleur.

However, if the shifters are unpaired and re-paired, `Payload2` will have a different value under the same counter conditions, while `Payload1` remains unchanged even after unpairing. So, in other words, `Payload2` is susceptible to changes upon reconfiguring the shifting system components.

Destination ID: The acknowledgment packets feature a 16-bit `Destination ID` at their conclusion. This ID corresponds to the identity of the shifter—the commanding device or the device that the acknowledgment is intended for. Through extensive testing involving various pairings of shifters and derailleurs, we consistently observed that the `Destination ID` is determined by the shifter that sends the command packet. In other words, the acknowledgment identifies and responds to the shifter initiating the command.

Additionally, our experiments revealed that the `Destination ID` remains constant, even after devices are unpaired and then repaired. This consistency indicates that the `Destination ID` is inherently linked to the shifter itself and does not change with different pairing configurations. It highlights that the identity encoded in the `Destination ID` is intrinsic to the shifter rather than being dependent on the pairing status or the particular session of interaction between the devices.

It is worth mentioning that the command packets lack a feature similar to the `Destination ID`, which is consistent and unencrypted, that would allow one to identify the receiver from captured messages.

For our analysis, we looked into the packets from shifters controlling both the rear and front derailleurs. We confirmed that the packet structure remains consistent for all command packets and constant across all acknowledgment packets.

3.4 Security Weakness

Our analysis revealed that Shimano's wireless gear shifting protocol employs a form of encryption, which hinders attackers from creating and transmitting their own packets to the

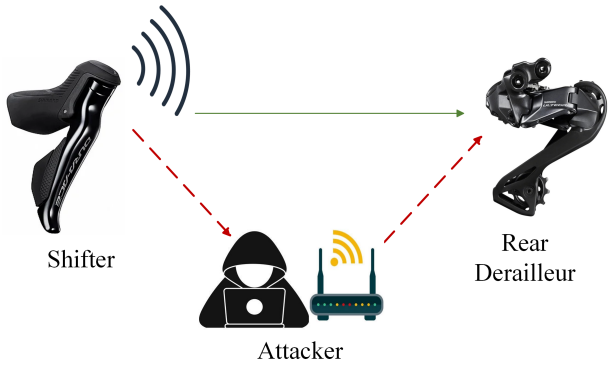

Figure 6: The attacker model for the replay attack. The attacker captures the command signal from the shifter, sends it to the derailleur, and later replays it.

derailleurs. However, the protocol does not offer protection against replay attacks, as the packets lack timestamps and sequence numbers, rendering the protocol vulnerable to such attacks. In this scenario, the attacker is not required to fabricate packets but can simply capture and replay them. We further describe this vulnerability with our experimental findings in the following section.

4 Replay and Jamming Attacks

In the following, we will explore the susceptibility of Shimano's wireless gear shifting protocol to replay attacks in Section 4.2, and discuss targeted jamming against this protocol in Section 4.3.

4.1 Attacker Assumptions and Experimental Setup

The attacker is equipped with an SDR capable of transmitting and receiving signals in the 2.4 GHz band. All commercial off-the-shelf SDRs, such as the USRP B210 [5], HackRF [7], PlutoSDR [11], and LimeSDR [9], are potential options for this purpose. In our experiments, we used an USRP B210. While the attacker may opt for more advanced setup, e.g., amplifiers to extend the attack range, these are not essential components in our baseline attacker model.

A replay attack in RF communications is when an attacker captures the legitimate signals and retransmits them to execute authenticated actions on a system without authorization. This vulnerability poses significant risks as it can bypass various security measures, including data encryption. To perform a successful replay attack, the attacker does not need to know the packet's format or contents. Replay attacks can even work against systems with encrypted protocols. In our targeted jamming attack, we capture and retransmit the signal similar to

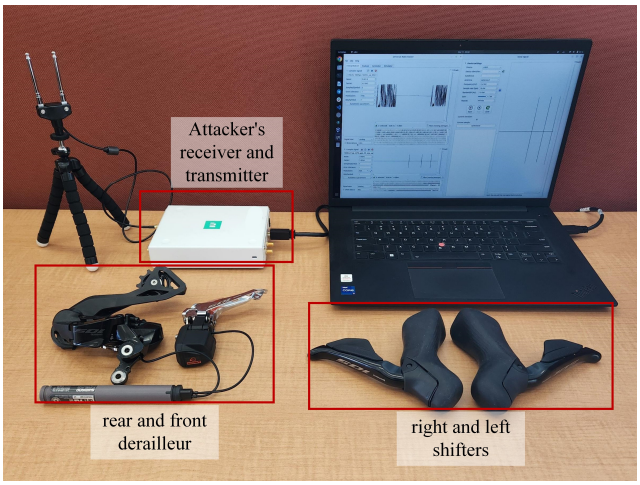

Figure 7: Our experimental setup, featuring the Shimano wireless gear shifting system (including shifters and derailleurs) and the USRP B210 as the attacker's transceiver.

the replay attack. The methodology and details are explained in Section 4.3.

As previously mentioned, the attacker is equipped with an SDR; for our experiments, we used a USRP B210 to transmit and receive signals without external amplifiers. Figure 6 depicts the attack model. The attacker's strategy involves capturing the signal emitted when the user engages the button to shift gears up or down. Once captured, replaying this signal enables gear shifting on the target's bike, The attack works independently of the system's current gear, effectively allowing for unauthorized control over gear adjustments. Figure 7 shows a photo of our evaluation setup.

A pre-requisite is that the attacker can capture a single upshift and downshift signal. There are several situations in which the attacker could collect these transmissions. An attacker does not need physical access to the bike; being in the vicinity is sufficient to capture the signal remotely in just a matter of seconds. For example, at a professional race, many individuals are within close proximity to a racer's bike. The attacker can capture the signals on the fly as the victim rider is actually shifting their bike's gears. Recall that our attack works irrespective of which gear the bike is currently in; thus, it is sufficient for the attacker to capture *any* upshift and *any* downshift signal. In a professional race, the attacker is easily within the signal range of the victim rider (e.g., riders are just a few feet apart).

4.2 Replay attack

We explored the mechanics and implications of replay attacks within the context of Shimano's Wireless Gear Control Protocol. We detail our experimental setup and methodology using SDRs to transmit and receive signals, demonstrating how an attacker can exploit the system without needing to decrypt or

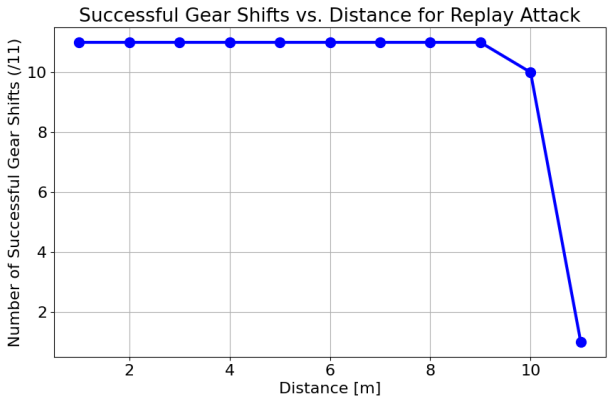

Figure 8: Assessing the Effective Distance for Replay Attacks without amplifier. Replay attack success rate vs distance from the target system.

even understand the signal's content.

Figure 8 presents the outcome of our replay attack experiments. The distance is measured from the attacker's transmitter to the bike's rear derailleur. At various distances, we conducted tests to shift the gear from the lowest to the highest level, encompassing eleven levels in total. Our results indicate that the replay attack is effective up to a distance of 9 meters without encountering any failures. At a distance of 10 meters, we observed an average success rate of 10 out of 11 attempted gear shifts. Each test involved shifting through all 11 gears, from the lowest to the highest. Beyond 10 meters, the signal falls outside the effective range. Consequently, for the attack to be viable, the attacker must be within 10 meters of the target bike. All tests were conducted multiple times to ensure reliability. Specifically, each test was repeated at least five times across all shift levels. Additionally, for critical aspects of our study, such as measurements beyond 10 meters, we increased the number of repetitions to up to ten times to confirm the protocol's effective range.

A critical point is that the attacker does not require direct physical access to the bike to capture and store the necessary signal. Once recorded, these signals can be reused at any future point without issue, owing to the lack of timestamps in the packet data.

The system completely lacks defenses against replay attacks, a finding reinforced by our ability to successfully replay the same signals two months after initially capturing the packets. Additionally, we conducted an experiment in which we recorded and replayed the signal, manually made at least 400 shifts, and subsequently performed the replay again, which proved to be effective. As long as the shifters remain paired with the same derailleur, the captured packets remain effective for replay.

Furthermore, by capturing just one instance of upshifting and one of downshifting from the targeted bike, an attacker can create any sequence of gear shifts at varying intervals. This enables them to carry out attacks at any future point as long as the derailleur remains paired with the same shifters. We successfully created and executed our arbitrary sequences of upshifting and downshifting through replay attacks. In summary, this has the effect of creating an unauthorized shifter that completely controls the rear derailleur and the front derailleur of the victim. We successfully replicated the experiments by replaying the control commands (upshifting and downshifting) for the front derailleur and managed to take control of it.

4.3 RF Jamming Attack

A jammer operates as an RF transmitter, transmitting noise that interferes with wireless communications. Our study utilized two varieties of jammers: one generating random noise and another broadcasting sine and cosine waves. To assess the effectiveness of our jammers, we carried out tests under various conditions. Based on our observations, using sine and cosine waves for the jamming signal proved more effective than noise. Consequently, we conducted our experiments to assess the jamming range by generating sine and cosine waves.

We transmitted the generated signal precisely at 2.478 GHz to interfere with the communication, as this is the specific frequency used for all Shimano proprietary communications. Consequently, the jamming would affect all nearby bikes operating on this frequency. In our jamming tests, we positioned the shifter and the derailleur one meter apart, reflecting the typical distance between these components on a bike. We then experimented with the jammer at varying distances.

We use GNURadio [6] for generating our jamming signals and a USRP B210 to transmit them over the air.[2] The effectiveness of jamming depends on various factors, including the power of the jamming device, the type of signal being jammed, the environmental conditions, and the distance between the jammer and the receiver. The result of jamming can vary significantly based on these conditions. If the jammer is located anywhere within the one meter zone from the derailleur, the gear-shifting system becomes completely non-functional, losing all capability for successful communication.

Generally, jamming effectiveness increases as the jammer gets closer to its target. Outside the tested ideal jammer zone (1 meter from the derailleur), the jammer still disrupts communication to some extent, but it doesn't completely disable the bike's functionality. Our jamming range experiment was conducted using a baseline setup without the enhancement of amplifiers or directional antennas. There are multiple methods to make jamming more effective. Directional antennas, for instance, could intensify the jamming signal's focus toward a particular area while lessening its effect elsewhere. The

[2]We note that although the devices operate in ISM band, care was taken to isolate the experimental setup in a separate area.

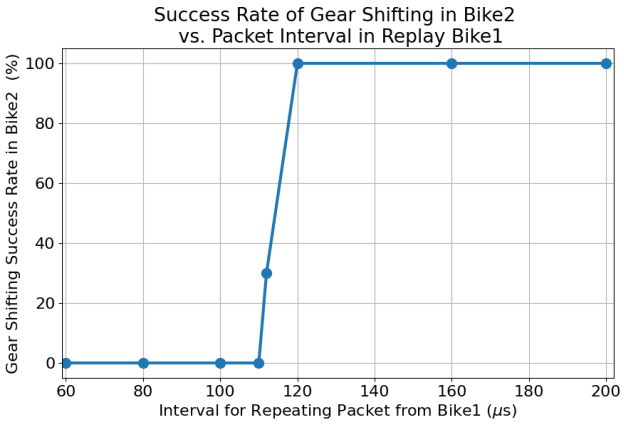

Figure 9: Analysis of Gear Shifting Success Rate in Bike2 Relative to Packet Interval Timing from Bike1: Demonstrating the Impact of Interference on Functionality

operational success and strategy of jamming can be greatly influenced by directionality, based on the jammer's construction and strategic goals.

In a targeted jamming attack, the attacker would try to replay the signal on one bike so that it doesn't cause any interference to other bikes. In our study, we labeled two Shimano wireless gear-shifting sets as Bike1 and Bike2. We captured an upshifting signal from Bike1 and replayed it at various intervals using a USRP B210. The targeted bike suddenly goes to the highest gear and stops there. Simultaneously, we attempted manual gear shifting on Bike2 in the vicinity. Our findings, illustrated in Figure 9, reveal that if the interval is less than 112 µs, which is equal to one packet length, Bike2 also stops working due to interference. The interference on Bike2 ceases when replay intervals exceed 112 µs, allowing normal communication due to sufficient time for transmitting command packets and receiving acknowledgments. In conclusion, when the attacker sends the replay packets with 112 µs interval, Bike1 ceases to function, whereas Bike2 or any other bike continues to operate normally.

5 Eavesdropping ANT Communication

Shimano utilizes the ANT+ protocol to transmit data, which devices like cycle computers can then pick up and display for cyclists. It's important to note that the Shimano wireless gear shifting system does not send control commands via the ANT+ protocol. In ANT+ communication, multiple devices can connect to a single source. This means that with the Shimano network key, any nearby ANT+ receiver can pick up the data being transmitted. For instance, if two bikes are close together, the second bike can link to the first bike's transmission and access its data simultaneously as the first bike is connected to its cycle computer. This allows an attacker to time their gear shifting replay based on precise knowledge of what gear the victim rider is using.

Table 7: Configuration for Capturing Shimano's ANT+ communication

Channel Frequency	57 (0x39), 2457 MHz
Network Key	A9-AD-32-68-3D-76-C7-4D
Channel Type	Master (0x10), Slave (0x00)
Device Number	1-65535, 0 searching
Device Type	1 (0x01)
Transmission Type	5 (0x05)
Channel Period	8198 counts, 4 Hz

We emphasize that eavesdropping on ANT+ communication does not form a core component of our attacker model. However, being able to target a specific gear through eavesdropping can indeed offer a strategic advantage to an attacker.

Table 7 outlines the configuration parameters necessary for capturing data using Shimano's ANT+ protocol. The critical piece of information is the Shimano network key, a unique identifier that secures and enables communication on the Shimano ANT+ network. Cycle computers need this network key to capture ANT+ communications from Shimano devices. The Channel Frequency identifies the specific radio frequency used for communication, with channel 57 operating at 2457 MHz, which helps avoid signal interference. The Channel Type indicates whether a device acts as a 'Master' (initiating communication) or a 'Slave' (receiving data), with specific hexadecimal values for setup. The Device Number serves as a unique identifier within the ANT+ network. Numbers range from 1 to 65535, with 0 reserved for searching mode to connect with nearby devices. Device Type corresponds to the ANT+ standard for different device categories, with a type of 1 usually denoting a generic sensor. Transmission Type refers to specific patterns or information for device communication. The Channel Period details the frequency of data broadcasts, with '8198 counts' equating to a 4 Hz rate, affecting both data timeliness and device battery life. Using the ANTware II [2] application and the correct configuration, we managed to intercept communications on the Shimano ANT+ network. ANTware II is a tool for managing ANT/ANT+ devices via an ANT+ USB stick. Figure 10 displays the captured packets during gear shifts from 8 to 1 on Shimano's ANT+ network. The data highlighted in red represent the current gear values. Having a detailed knowledge of the network parameters and packet structure allows an attacker to easily replicate these packets.

6 Discussion

Shimano's protocol incorporates basic encryption techniques to prevent attackers from creating counterfeit signals. Reverse engineering was notably demanding due to its use of GFSK modulation, which complicates demodulation when parame-

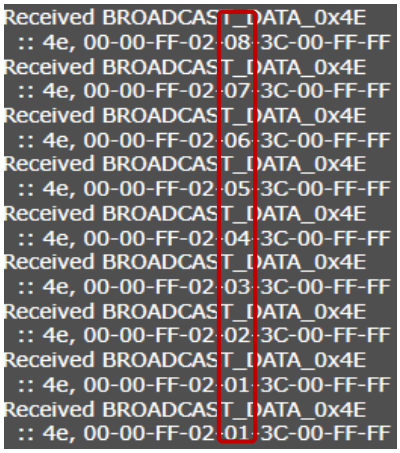

Figure 10: Eavesdropping ANT communication. The data highlighted in red represent the current gear values in each packet.

ters are undisclosed. This is very different from the amplitude shift keying used by numerous security systems, which, due to its simplicity, leaves them more open to security breaches. In the current landscape, where many wireless devices communicate without encryption, exposing them to security threats, Shimano stands out by implementing encrypted communication, enhancing its defense against direct hacking. However, the system remains exposed to replay attacks. Below, we outline recommended strategies to mitigate the risk of replay attacks.

Effects of the Attacks. A modern bicycle typically has two derailleurs that control the chain position: rear and front. The rear derailleur typically has 11 or 12 levels, and the changes between levels are usually minor but still impact the rider's performance. The front derailleur has two levels with large gear ratios. For example, consider a racer who is climbing a mountain. They will typically be in the smallest gear on the front. If the attack targets the front derailleur, causing it to move into a larger gear (i.e., harder for the rider), it can significantly impact rider performance, force them to stop, or even snap the chain. In professional races, any unintended changes to the gear position will have drastic consequences and affect the integrity of the sport. We believe that unauthorized gear changes through the attacks highlighted in our paper have a similar effect on the sport as performance-enhancing illegal drugs.

Size/Cost of Attack Device. In the current implementation of our signal capture and replay system, we utilize a setup comprising a SDR and a laptop. While effective, this configuration is not optimized for size or portability. However, with advancements in miniaturization and integrated circuit (IC) technology, it is feasible to reduce the size of the attack device significantly. By custom designing specific circuits, we can integrate a receiver, a modest amount of memory for signal storage, and a transmitter into a compact, single System on a Chip (SoC) or small circuit board. This miniaturization process makes the attack system more discreet and enhances its portability and deployment ease. For example, researchers demonstrated relay attacks [20] on passive keyless entry systems with SDRs costing more than $1500 in 2011. A few years later, the same attack was demonstrated using $225 [12].

Countermeasures. Adding timestamps into wireless communications can mitigate replay attacks to some degree by allowing only messages sent within a designated timeframe, thereby rendering older, possibly replayed messages invalid. Nonetheless, integrating timestamps into wireless communication poses challenges. Effective use of timestamps requires precise synchronization between the devices. This can be challenging, particularly in settings where devices lack consistent access to a shared time source, such as the Internet or GPS signals.

Rolling or hopping codes stand as another prevalent strategy in wireless systems to prevent replay attacks. Within this framework, each transmitted signal is accompanied by a distinct code generated through a specific algorithm known to both the sender and receiver. These codes are one-time-use only, ensuring that once a code has been utilized for authentication, it is voided, prompting both devices to proceed to the subsequent sequence code. This method is especially prevalent in scenarios prone to signal interception and unauthorized reuse, such as passive keyless entry in cars and garage door systems. Although rolling codes significantly counter basic replay attacks, they are not foolproof against more sophisticated threats, such as code grabbing and delayed playback if an attacker intercepts the original code's delivery to the receiver. However, this approach can significantly increase the difficulty of performing a replay attack in Shimano wireless gear shifting.

There are other types of countermeasures designed for specific applications that can be highly effective and useful. Particularly for Shimano bikes, implementing distance-based restrictions could be beneficial [35]. Since legitimate interactions occur only between shifters and derailleurs within limited distances, establishing range limitations on the receiver to only accept commands from close proximity can be helpful. This approach is based on the assumption that attackers are more likely to conduct replay attacks from a distance, so by restricting the range at which commands are accepted, we reduce the likelihood of successful remote attacks. However, securely measuring distance is a challenging problem [33, 36] in itself, and therefore, while it can reduce the risk of replay attacks, it should be used in conjunction with other security measures for comprehensive protection.

Our current observation indicates that it is likely that Shimano is not using any kind of rolling code or other mentioned

countermeasures. Our study reveals that the current security measures in Shimano's wireless gear shifting systems are insufficient to protect against replay attacks. The practical feasibility of executing these identified attacks demonstrates that attackers could exploit these vulnerabilities with relatively modest resources. Despite advancements in similar systems, the lessons learned have not been fully integrated into the design of Shimano's wireless gear shifters, leaving them vulnerable to specific cyber attacks such as those observed in automotive key fob systems. Moving forward, it is crucial to implement robust defenses, including rolling codes and other complementary security measures, to enhance the security guarantees of these wireless systems and safeguard against potential attacks.

E-TUBE PROJECT. The Shimano E-TUBE PROJECT is a platform that allows users to customize, update, and diagnose Shimano's electronic gear-shifting systems via BLE. When connected to the E-TUBE PROJECT over BLE, the rear derailleur would be out of operation. If the malicious attacker has any chance to physically access the derailleur, they can easily pair their phone with it and cause a DoS (Denial of service). The biker in this situation would not know what is wrong, and the only way to fix it is to completely disconnect the derailleur from the battery to cut off the power. Users are strongly advised to change the default passkey immediately after acquisition. Often, the initial pairing occurs at dealerships, which may result in the default code remaining unchanged if the end-user is not prompted to modify it. Furthermore, enhancing BLE security with unique, secure passkeys for each bike, rather than a standard default passkey, is recommended to prevent unauthorized access. If an attacker manages to connect the bike to his E-TUBE PROJECT, the implications can be severe. They could easily alter the bike's settings and even change the passkey, preventing any quick fix.

Future Work. In future research, we intend to expand our investigation into the security architecture of wireless gear-shifting systems beyond the scope of Shimano. We plan to analyze and compare various manufacturers' vulnerabilities and defense mechanisms, identifying common weaknesses and best practices within the industry. This comprehensive analysis will allow us to develop more robust security guidelines and recommendations for all wireless gear-shifting systems. Our goal is to ensure safer and more secure cycling experiences for users.

7 Related Work

In this section, we will review two primary areas of focus related to our work. First, we examine previous research on the reverse engineering of wireless systems. Second, we describe the security challenges posed by replay and relay attacks, exploring how these threats impact various technological domains.

7.1 Analysis of Proprietary Wireless Protocols

Many devices contain inherent security flaws. They often rely on security through obscurity by keeping their protocols and information secret, hoping this discourages efforts to reverse engineer and uncover potential vulnerabilities.

Various reverse-engineering studies have been conducted on different devices [18, 40], each with unique attributes and methodologies. For example, Garcia et al. [22] investigated the security of wireless smart cards used in payment systems, while Strobel et al. [39] examined a digital locking and access control system prevalent in corporations and educational institutions. Both studies required physically opening the devices to connect the wireless chips to a logic analyzer, which is invasive and could be easily detected compared to non-invasive techniques. Contrastingly, non-invasive reverse engineering, such as intercepting wireless communications using SDRs, offers a less detectable, scalable, and repeatable approach. This method avoids the complications of hardware tampering while still providing deep insights into wireless protocols. For example, [32] research on wireless mice and keyboards, which often use proprietary protocols in the 2.4 GHz ISM band.

Kim et al. [26] report instances in which authors could eavesdrop by recovering the 128-bit AES key. In [27], the process of demodulating RF signals into binary data for analysis was documented for a smart home alarm system known as SecuritasHome.

Researchers have recently adopted hybrid approaches for reverse engineering and launching attacks. Notable instances include Samy Kamkar's innovative methods for remote keyless entry systems [25] and Mike Ryan et al. [37] for electric skateboard control interfaces. Also, in [23], the authors focused on a case study with rolling codes.

Tools like URH [34] have aimed to streamline the reverse engineering process of wireless protocols, offering an open-source solution for signal capture and protocol analysis through SDRs. RFQuack [29] represents another advancement, a modular RF dongle system that allows for the customized development of dongles tailored to specific reverse engineering needs in wireless protocols. This tool underscores the evolving landscape of non-invasive techniques in security research.

In addition to academic studies, there have been non-academic reverse engineering efforts on the Shimano Di2 system [13, 14, 30]. However, these efforts primarily focus on reverse engineering the ANT communication protocol. To the best of our knowledge, none of these works have explored Shimano's proprietary protocol. Furthermore, none have investigated replay attacks or targeted jamming attacks on Shimano's command signals.

7.2 Replay and Relay Attack

Replay and relay attacks pose significant threats in wireless communications. It enables attackers to capture and rebroadcast packets for unauthorized access or service disruption, impacting various systems such as keyless vehicle entry, GPS, and remote garage door openers. For instance, previously, researchers have shown that through a relay attack, where a device is used to extend the communication between two legitimate devices, it's possible to unlock a vehicle and drive away even when the actual key is far from the car [20].

Similarly, GPS spoofing mirrors these concerns, with studies like [24, 31] demonstrating the potential for GPS signal manipulation, impacting navigation and timing. In RF communication, Roland et al. [21] explore relay attack risks in NFC transactions, commonly used in touchless payment and entry systems. The challenge in the abovementioned works would be relaying the signal in real-time. However, as shown in this paper, our attack on the wireless gear shifting system doesn't necessitate real-time relays and can be executed using any packet previously captured. RFID systems, crucial for secure access and transactions, face similar threats, with [38] addressing these system's susceptibilities to replay attacks.

Additionally, the increase in replay attacks on smart home systems underscores growing security gaps, as examined by Fernandes et al. [19], spotlighting exploitable weaknesses in smart home protocols.

8 Conclusion

The sport of cycling is an adversarial environment. Modern bicycles are cyber-physical systems that support wireless control of gear shifting. We conducted the first security analysis of the Shimano wireless shifting protocol and discovered its vulnerability to replay and jamming attacks. This allows attackers to target riders and take over control of the bike's gear shifting behavior. Allowing attackers such control can lead to negative outcomes on the performance of riders in professional races and can affect the integrity of the sport.

We discussed our analysis of Shimano's protocol with the hope that it would bring additional scrutiny to these technologies. We envision that future work will investigate the security of other wireless gear control manufacturers. Long term, we outlined countermeasures that manufacturers could use to reduce the impact of attacks. For example, a rolling codes system can reduce the attacker's ability to arbitrarily control gear changes.

Acknowledgements

The work was partially supported by NSF grant 2144914. We thank Keith Wakeham and Virgyl Fernandes for their technical expertise in cycling components, Andreas Noack for his expert suggestions on URH, Yoshi Kohno for early discussions, the anonymous reviewers and our shepherd, Manuel Egele, for their insightful comments, and finally Stefan Savage, Geoff Voelker and the UCSD SysNet group for paper title ideas.

References

[1] ANT / ANT+ Defined. https://www.thisisant.com/developer/ant-plus/ant-antplus-defined.

[2] ANTware II. https://www.thisisant.com/developer/resources/software-tools.

[3] Campagnolo. https://www.campagnolo.com/.

[4] E-TUBE PROJECT. https://bike.shimano.com/en-US/e-tube.html.

[5] Ettus Research. https://www.ettus.com/products/.

[6] GNU Radio. https://www.gnuradio.org/.

[7] HackRF. https://greatscottgadgets.com/hackrf/one/.

[8] Inspectrum. https://github.com/miek/inspectrum.

[9] LimeSDR. https://limemicro.com/products/boards/limesdr/.

[10] New shimano 105 di2 delivers 12-speed, shifting and wireless performance. https://bike.shimano.com/en-US/information/news/new-shimano-105-di2-delivers-12-speed–shifting-and-wireless-per.html.

[11] PlutoSDR. https://wiki.analog.com/university/tools/pluto.

[12] Radio Attack Lets Hackers Steal 24 Different Car Models. https://www.wired.com/2016/03/study-finds-24-car-models-open-unlocking-ignition-hack/.

[13] REVERSE ENGINEERING SHIMANO BIKE ELECTRONICS. https://hackaday.com/2019/03/26/reverse-engineering-shimano-bike-electronics/.

[14] Reverse Engineering Shimano DI2. https://titanlab.co/reverse-engineering-shimano-di2/.

[15] Shimano: Bike component monopoly at an attractive price. https://seekingalpha.com/article/4572435-shimano-bike-component-monopoly-at-an-attractive-price.

[16] Shimano dura-ace di2. https://bike.shimano.com/en-EU/product/component/dura-ace-r9200.html.

[17] SRAM. https://www.sram.com/.

[18] CHOI, K., SON, Y., NOH, J., SHIN, H., CHOI, J., AND KIM, Y. Dissecting customized protocols: automatic analysis for customized protocols based on ieee 802.15. 4. In *Proceedings of the 9th ACM Conference on Security & Privacy in Wireless and Mobile Networks* (2016), pp. 183–193.

[19] FERNANDES, E., JUNG, J., AND PRAKASH, A. Security analysis of emerging smart home applications. In *2016 IEEE symposium on security and privacy (SP)* (2016), IEEE, pp. 636–654.

[20] FRANCILLON, A., DANEV, B., AND CAPKUN, S. Relay attacks on passive keyless entry and start systems in modern cars. In *Proceedings of the Network and Distributed System Security Symposium (NDSS)* (2011), Eidgenössische Technische Hochschule Zürich, Department of Computer Science.

[21] FRANCIS, L., HANCKE, G., MAYES, K., AND MARKANTONAKIS, K. Practical relay attack on contactless transactions by using nfc mobile phones. *Cryptology ePrint Archive* (2011).

[22] GARCIA, F. D., DE KONING GANS, G., MUIJRERS, R., VAN ROSSUM, P., VERDULT, R., SCHREUR, R. W., AND JACOBS, B. Dismantling mifare classic. In *Computer Security-ESORICS 2008: 13th European Symposium on Research in Computer Security, Málaga, Spain, October 6-8, 2008. Proceedings 13* (2008), Springer, pp. 97–114.

[23] GHANEM, A., AND ALTAWY, R. Garage door openers: A rolling code protocol case study. In *2022 19th Annual International Conference on Privacy, Security & Trust (PST)* (2022), IEEE, pp. 1–6.

[24] HUMPHREYS, T. E., LEDVINA, B. M., PSIAKI, M. L., O'HANLON, B. W., KINTNER, P. M., ET AL. Assessing the spoofing threat: Development of a portable gps civilian spoofer. In *Proceedings of the 21st International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2008)* (2008), pp. 2314–2325.

[25] KAMKAR, S. Keysweeper, 2015.

[26] KIM, K., KIM, T. H., KIM, T., AND RYU, S. Aes wireless keyboard: Template attack for eavesdropping. *Black Hat Asia, Singapore* (2018).

[27] LINDEBERG, A. Hacking into someone's home using radio waves: Ethical hacking of securitas' alarm system, 2021.

[28] LLOYD, D. The fight for the crown: Shimano vs sram, 2018. https://www.bicycling.co.za/bikes-gear/the-fight-for-the-crown-shimano-vs-sram/.

[29] MAGGI, F., AND GUGLIELMINI, A. Rfquack: A universal hardware-software toolkit for wireless protocol (security) analysis and research. *arXiv preprint arXiv:2104.02551* (2021).

[30] MILLER, S. Shimano Di2 Security. https://www.youtube.com/watch?v=Kxecl08qB60&ab$_{c}$*hannel* = *ShaneMiller* − *GPLama*.

[31] MOTALLEBIGHOMI, M., SATHAYE, H., SINGH, M., AND RANGANATHAN, A. Location-independent gnss relay attacks: A lazy attacker's guide to bypassing navigation message authentication. In *Proceedings of the 16th ACM Conference on Security and Privacy in Wireless and Mobile Networks* (2023), pp. 365–376.

[32] NEWLIN, M. Mousejack: Injecting keystrokes into wireless mice. *Retrieved January 10* (2016), 2019.

[33] ÓLAFSDÓTTIR, H., RANGANATHAN, A., AND CAPKUN, S. On the security of carrier phase-based ranging. In *International Conference on Cryptographic Hardware and Embedded Systems* (2017), Springer, pp. 490–509.

[34] POHL, J., AND NOACK, A. Universal radio hacker: A suite for analyzing and attacking stateful wireless protocols. In *12th USENIX Workshop on Offensive Technologies (WOOT 18)* (2018).

[35] RANGANATHAN, A., AND CAPKUN, S. Are we really close? verifying proximity in wireless systems. *IEEE Security & Privacy 15*, 3 (2017), 52–58.

[36] RANGANATHAN, A., DANEV, B., FRANCILLON, A., AND CAPKUN, S. Physical-layer attacks on chirp-based ranging systems. In *Proceedings of the fifth ACM conference on Security and Privacy in Wireless and Mobile Networks* (2012), pp. 15–26.

[37] RYAN, M., AND HEALEY, R. Hacking electric skateboards: Vehicle research for mortals, 2015. https://www.defcon.org/html/defcon-23/dc-23-speakers.html.

[38] SINGH, A. K., AND PATRO, B. Security attacks on rfid and their countermeasures. In *Computer Communication, Networking and IoT: Proceedings of ICICC 2020* (2021), Springer, pp. 509–518.

[39] STROBEL, D., DRIESSEN, B., KASPER, T., LEANDER, G., OSWALD, D., SCHELLENBERG, F., AND PAAR, C. Fuming acid and cryptanalysis: Handy tools for overcoming a digital locking and access control system. In *Advances in Cryptology–CRYPTO 2013: 33rd Annual Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2013. Proceedings, Part I* (2013), Springer, pp. 147–164.

[40] STUTE, M., KREITSCHMANN, D., AND HOLLICK, M. Reverse engineering and evaluating the apple wireless direct link protocol. *GetMobile: Mobile Computing and Communications 23*, 1 (2019), 30–33.

[41] ZHANG, Y., WENG, J., DEY, R., JIN, Y., LIN, Z., AND FU, X. Breaking secure pairing of bluetooth low energy using downgrade attacks. In *29th USENIX Security Symposium (USENIX Security 20)* (2020), pp. 37–54.