PuppetMouse: Practical and Contactless Mouse Manipulation Attack via Intentional Electromagnetic Interference Injection

WENFAN SONG, Zhejiang University, China
JIANWEI LIU, Zhejiang University, China, and Hangzhou City University, China
JINSONG HAN*, Zhejiang University, China

Mouse is a ubiquitous input tool crucial for user-computer interaction in modern society. However, the inherent trust in the mouse may pose security risks. If the mouse is maliciously manipulated, the connected computer could be secretly controlled, endangering personal privacy and property. In this paper, we introduce PuppetMouse, the first intentional electromagnetic interference (IEMI) injection-based attack that can effectively manipulate mouse clicks and movements without physical contact. By adjusting the parameters of IEMI signals, PuppetMouse can precisely control the click side, as well as the movement direction and speed. We demonstrate PuppetMouse's effectiveness on 14 wired and wireless mice from popular brands. The short response delay (within 4 ms) affirms the real-time performance of PuppetMouse. Robustness analysis across different attack distances and material occlusions validate the stability and reliability of PuppetMouse. Two case studies on firewall disabling and malicious WiFi connection further prove the severe threats of PuppetMouse in the real world. We also propose an integrated set of hardware and software-based defensive mechanisms to mitigate the risks posed by PuppetMouse.

CCS Concepts: • **Security and privacy** → *Hardware attacks and countermeasures.*

Additional Key Words and Phrases: Hardware Security, Intentional Electromagnetic Interference Injection, Mice

ACM Reference Format:
Wenfan Song, Jianwei Liu, and Jinsong Han. 2024. PuppetMouse: Practical and Contactless Mouse Manipulation Attack via Intentional Electromagnetic Interference Injection. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 8, 3, Article 125 (September 2024), 30 pages. https://doi.org/10.1145/3678570

1 INTRODUCTION

Mouse is an indispensable input tool in modern information society, which plays a crucial role in user-computer interaction. The operations of the mouse, e.g., clicking and moving, enable users to perform various tasks on the computer, such as browsing the webpage and opening files. Nowadays, optical mice (abbr. mice) dominate the mouse market [55] due to their durability, high precision, and sensitivity. They have been extensively employed on computer systems (e.g., hosts and laptops) and even safety-critical systems like industrial control systems [2] and medical equipment [12].

Nevertheless, as a ubiquitous and inherently trusted input peripheral, the interaction between mice and computers typically does not necessitate stringent security credentials. In this case, once the mouse is maliciously manipulated, its connected computer and the user may be subjected to substantial risks, primarily evident in the

*Jinsong Han is the corresponding author.

Authors' addresses: Wenfan Song, Zhejiang University, China, wenfansong@zju.edu.cn; Jianwei Liu, Zhejiang University, China, and Hangzhou City University, China, jianweiliu@zju.edu.cn; Jinsong Han, Zhejiang University, China, hanjinsong@zju.edu.cn.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
© 2024 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM 2474-9567/2024/9-ART125
https://doi.org/10.1145/3678570

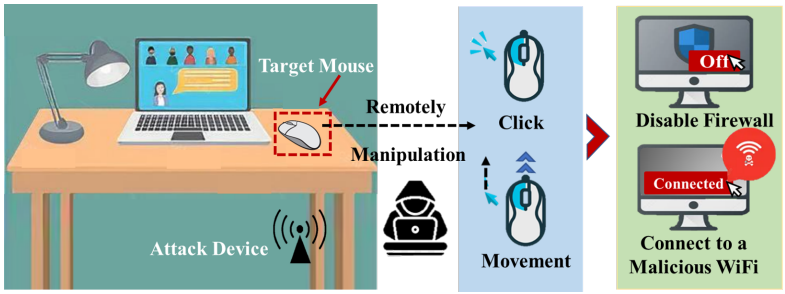


Fig. 1. By injecting IEMI signals to the target mouse, an attacker can remotely manipulate its click and movement behaviors to perform a series of malicious actions on its connected computer (e.g., a laptop).

following aspects: (1) Malicious takeover of host. Attackers can simulate legitimate user actions to execute various malicious activities, including downloading and installing malware [8, 15, 48], modifying system settings [19], or tampering with data [43]. This manipulation poses a severe threat to the integrity and security of the host system. (2) User privacy breach. Attackers can exploit controlled privileges to access and steal sensitive user data [9, 34, 35], such as personal profiles, financial information, and login credentials. This information can be used for identity theft, financial fraud, and other illegal activities, resulting in significant personal and financial losses to users. (3) Remote hijacking of critical infrastructure. By controlling the mouse, attackers can remotely hijack critical infrastructure connected to the host, such as water supply systems [18], and financial systems [32]. Such remote hijacking could lead to significant societal impacts, endangering people's lives and property.

Existing attacks targeting mice can be categorized into three main types: computer-side attacks [9, 15, 43], computer-mouse communication attacks [1, 34, 35, 42, 58], and mouse-side attacks [8, 19, 28, 48]. (1) Computer-side attacks typically use malware to maliciously hijack the mouse [15, 43]or monitor the mouse [9]. However, these malware-based methods require invading the computer system, resulting in notable overhead. Moreover, abnormal behaviors caused by malware are prone to be detected by users or security software. (2) The second type of attacks are at the protocol level. They involve sniffing [34, 35] or tampering with [1, 42, 58] the data packets of mouse-computer communication to monitor or control the mouse's operations. Yet, protocol-based attacks lack universality as they are only applicable to specific protocols. (3) Attackers may also attempt to modify the firmware [8, 19, 48] or hardware [28] of the mouse to spread malicious code to the connected computer. Nevertheless, such attacks require physical contact or even full control of the mouse to carry out intrusive modifications. Meanwhile, these attacks are easy to raise user awareness.

In this paper, we consider a non-intrusive, universal, and stealthy attack approach targeting mice: injecting intentional electromagnetic interference (IEMI) signals. We introduce PuppetMouse, the first IEMI-based mouse attack that achieves fine-grained control of mouse clicks and movements without physical contact. The core idea is that the common mouse sensors (i.e., microswitches and image sensors) can serve as "backdoors". They enable IEMI signals to modify the outputs of the mouse at the physical layer, resulting in malicious clicks and movements. Figure. 1 shows an attack scenario of PuppetMouse in which a person leaves his laptop equipped with a mouse on the table. A concealed attack device is embedded under the table beneath the mouse. The attacker remotely emits IEMI signals to manipulate the target mouse and further gains control over the laptop. This allows the attacker to perform various spiteful activities, e.g., disabling the laptop's firewall and connecting to a malicious WiFi, posing severe threats to the user's safety of life and property.

To implement PuppetMouse in practice, we address the following questions: 1) **How can IEMI signals impact the mouse to induce clicks and movements?** To trigger effective mouse activities with IEMI injection,

we first perform an in-depth theoretical analysis of the interaction between the IEMI signal and two crucial components (microswitch and image sensor) responsible for mouse click and movement detection. Based on this investigation, we summarize the IEMI signal patterns that can cause effective clicks and movements in the physical world. We also conduct a series of preliminary experiments to confirm the validity of our theoretical derivations. 2) **How to precisely control the click side (left and right) and movement direction/speed with IEMI signals?** Since the microswitches detecting left- and right-side clicks bear the same working principle, we cannot determine which side of the click would be triggered with roughly generated IEMI signals. Meanwhile, the direction and speed of the movement are hard to adjust. To solve this problem, we quantify the effects of the IEMI signal characteristics (such as frequency and amplitude) on the microswitches and image sensor. For precise click activation, we first analyze the difference in the physical structure of the left and right microswitch circuits and then adjust the IEMI signal parameters accordingly to control the click side. To realize direction- and speed-adjustable movements, we first design an optimal ghost pattern and superimpose it onto the images captured by the image sensor. Then, we adjust the IEMI signal parameters to control the drift properties (i.e., direction and speed) of the ghost pattern, which further controls the direction and speed of the cursor movement. 3) **How to ensure the independence of the click and movement events?** Since the microswitch and image sensor are close to each other, emitted IEMI signals may affect them simultaneously, resulting in interference between click and movement attacks.To tackle this issue, we propose a click-movement interference elimination strategy. On one hand, we fine-tune the carrier frequency of the IEMI signal to guarantee that the signal only affects one of the two components. On the other hand, we optimize the IEMI modulation signal by adjusting its properties to ensure that only clicks are triggered without causing movements, or vice versa.

We conduct real-world attack experiments on 14 mainstream brands of wired and wireless mice. The results show that PuppetMouse is capable of achieving precise and controllable mouse clicks and movements across diverse mouse models. Meanwhile, the short response delay (within 4 ms) and low interference rate (within 4.5%) confirm the practicality of such attacks. We also conduct a series of robustness experiments to validate the stability and reliability of PuppetMouse against different attack distances and various materials of occlusions. In addition, we demonstrate the practicality of PuppetMouse with two case studies: disabling the firewall and connecting to a malicious WiFi. We hope these examples will raise awareness of the dangers posed by such IEMI attacks, encouraging manufacturers and users to adopt stronger protective measures. To safeguard against the potential risks posed by PuppetMouse, we propose an integrated set of hardware and software-based defensive mechanisms. In summary, our contributions are as follows:

- We propose PuppetMouse, the first non-intrusive and contactless attack targeting both wired and wireless mice that can manipulate mouse clicks and movements through IEMI injection. This study helps to shed light on the interference of IEMI signals on mouse operation to encourage appropriate countermeasures. A demo of PuppetMouse is available online[1].
- We explore the feasibility of IEMI injection attacks and give detailed guidance with theoretical analysis and preliminary experiments, further quantitatively modeling the relationship between IEMI and two key components (i.e., microswitches and image sensors) in mice responsible for click and movement. These models enable fine-grained manipulation of the mice, namely, two-side clicks and direction- and speed-adjustable movements.
- To guarantee the independence of each attack, we propose a click-movement interference elimination strategy to ensure the IEMI injection of the target component while isolating the impact of IEMI on the non-target component, thereby achieving separate controls of clicks and movements, respectively.
- We implement the prototype of PuppetMouse and evaluate it on 14 mainstream brands of wired and wireless mice. Experimental results demonstrate that PuppetMouse can effectively and precisely trigger two-side

[1]Video demo: https://files.catbox.moe/ou5xdw.mp4

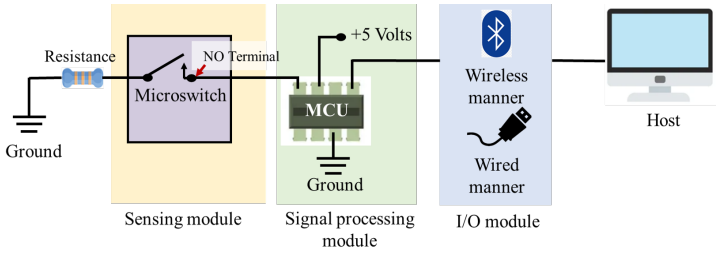


Fig. 2. Circuit underlying the click behaviour.

clicks and achieve direction- and speed-adjustable movements. Robustness analysis indicates that the attack effectiveness of PuppetMouse remains stable under different attack distances and material occlusions. Our case studies demonstrate the practicality of PuppetMouse in two attack scenarios: firewall disabling and malicious WiFi connection. Besides, we suggest hardware- and software-based countermeasures to mitigate such IEMI attacks.

2 BACKGROUND

Mouse is a widely adopted input peripheral that enables precise control of a computer (e.g., a host) by manipulating the on-screen cursor. In such interactions, click and movement are two principal functions. In this section, we respectively introduce the hardware basics and principles of mouse click and movement, paving the way for the subsequent malicious signal design.

2.1 Principle of Click

Figure 2 illustrates the circuit underlying the click behavior[2] which comprises three essential modules: a sensing module, a signal processing module, and an input/output (I/O) module. When the user presses the microswitch to trigger a click event, the microswitch converts the physical click behavior into an electrical signal (i.e., a switch signal V_{switch}). During this process, the normally open (NO) terminal of the microswitch changes from an open to a closed state. As a result, the voltage at the NO terminal (i.e., V_{NO}) shifts from a high-impedance state[3] (denoted as Z) to V_{switch}, expressed by:

$$V_{NO} = \begin{cases} Z, & \text{If microswitch is released.} \\ V_{switch}, & \text{If microswitch is pressed.} \end{cases} \tag{1}$$

The switch signal V_{switch} is then forwarded to the signal processing module, i.e., the micro-controller unit (MCU), which compares V_{switch} with a threshold V_{th} to confirm that the user has performed a click operation. At this point, V_{switch} meets the following criterion:

$$V_{switch} > V_{th}. \tag{2}$$

Afterwards, the mouse transmits the click command to the host via the I/O module in a wired (e.g., USB) or wireless (e.g., Bluetooth) manner. Finally, the host executes the click operation.

[2]All the microswitches on the mouse work in the same way.
[3]When V_{switch} is in a high-impedance state, the microswitch is not connected to the MCU and cannot input a valid voltage signal to the MCU.

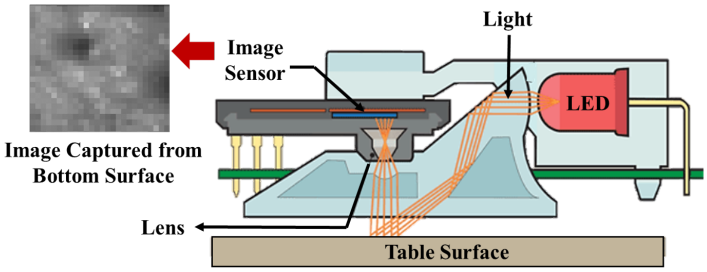


Fig. 3. Structure of IAS.

2.2 Principle of Movement

The movement of the mouse primarily relies on two components [56]: an image acquisition system (IAS) and a digital signal processor (DSP). The mouse first uses the IAS which functions like a miniature camera to continuously capture images of the surface beneath the mouse. Then, the DSP compares the differences between successive images to detect the mouse movement.

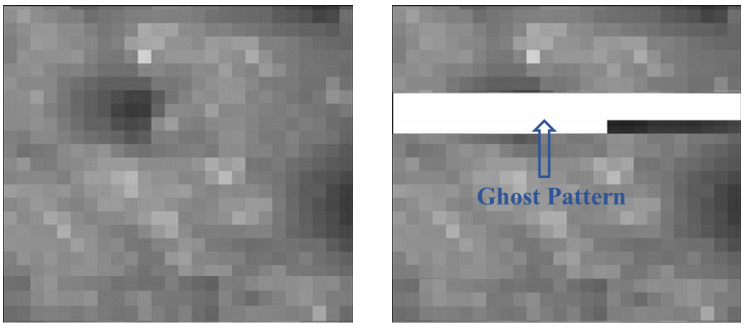
The IAS typically consists of an LED, a lens, and an image sensor [36], as shown in Fig. 3. During image acquisition, the LED first emits light to illuminate the surface beneath the mouse. Then, the light reflected from the surface travels through a series of components including a lens and a small aperture, to reach the image sensor. As shown in Fig. 3, the image sensor converts the incident light into images. Specifically, a 2D photodiode array within the image sensor absorbs the incident light and converts it into charges. The intensity of the light directly affects the amount of charge generated, with brighter areas generating more charge and darker ones generating less. Then, measuring units convert these charges into voltage, and the voltage is amplified by an amplifier. The amplified voltage signals are sequentially digitized by an analog-to-digital converter (ADC), starting from the position (0,0), and transformed into pixel values one by one. This sequence of "light→charge→voltage→pixel" conversion allows the IAS to continuously capture images of the surface beneath the mouse, which are then sent to the DSP for further data processing.

The DSP usually employs template matching algorithms [10] to compare the differences between adjacent images. Based on these differences, the DSP can determine whether the mouse is in movement and calculate the displacement of the movement. Assuming that there are two adjacent images, denoted as I^n and I^{n+1}. We first select a template T from the image I^n, and denote the top-left corner of T as (x_{ref}, y_{ref}). After that, the template T slides over the image I^{n+1} with a fixed step. At each sliding position (x, y), we extract a corresponding local region from I^{n+1} with the same shape as the template T, and then compute the similarity between this local region and T.The sliding position with the highest similarity is denoted as (x_{max}, y_{max}). By calculating the displacement between (x_{max}, y_{max}) and (x_{ref}, y_{ref}), we can obtain the mouse movement. Finally, the movement information is transmitted to the computer. The computer will update the cursor position on the screen accordingly.

3 THREAT MODEL

The goal of PuppetMouse attack is injecting well-crafted IEMI signals into a mouse to manipulate its behaviors (i.e., click and movement), with which the attacker can take charge of the target computer (i,e., a personal laptop) to perform a series of malicious activities, e.g., implant malware and connect to a malicious WiFi.

■ **Attacker's Capability.** We assume that the attacker is aware of the model of the victim mouse. This can be easily achieved by observing its appearance. Then, the attacker can acquire an identical model of the mouse from the market for study. Moreover, we assume that the attacker can generate and emit adversarial IEMI signals towards the mouse with a necessary device (i.e., a software-defined radio (SDR) like a USRP). The last common



(a) Benign image captured without IEMI injection.

(b) Image captured during IEMI injection. A ghost pattern is superimposed onto the benign image.

Fig. 4. Images captured by the image sensor without and with IEMI injection.

assumption is that the attacker can hide the attack device near the victim mouse (e.g., under the table where the mouse is placed) [17, 26, 51]. To make the attack more realistic, we also pose the following restrictions to the attacker: (1) S/he is not allowed to physically contact the victim mouse/its connected computer. So, s/he cannot modify the mouse/computer hardware or install any malware on them; (2) S/he cannot tamper with or intercept the communication between the mouse and the computer.

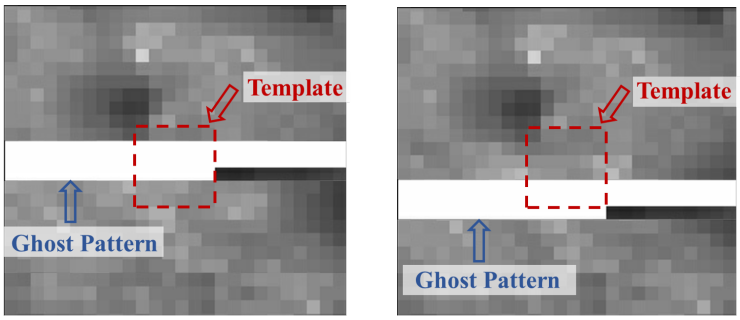
■ **Attack Scenarios.** PuppetMouse considers an attack scenario where a computer (e.g., a laptop) connected to a mouse is placed in a public area (such as a cafe or a meeting room). The mouse could belong to mainstream brands or models in the market [55], which usually employs microswitch and an image sensor to detect click and movement events, respectively. When the mouse is not manually operated by the user, the attacker can inject well-crafted IEMI signals into the mouse through the attack device hidden under the table, and precisely manipulate the click and movement behaviours of the mouse. This vulnerability allows attackers to gain complete control over the computer, which enables them to carry out a series of malicious activities. These actions include, but are not limited to, disabling firewalls, installing malware, and connecting to malicious wireless devices. These attacks not only pose a severe threat to the security of the affected computer but could also lead to even graver consequences. For instance, there could be sensitive data breaches, identity theft, or other forms of severe privacy violations, presenting users with significant security threats and chaos.

4 IEMI ATTACK FEASIBILITY STUDY

The feasibility of inducing mouse clicks and movements by IEMI is confirmed through a combination of theoretical derivation and experimental validation. First, we theoretically analyze the mechanisms of fake clicks (Sec. 4.1) and fake movements (Sec. 4.2) induced by IEMI signals. Then, we perform experiments to demonstrate the validity of these IEMI attacks in the physical world (Sec. 4.3).

4.1 Fake Click Triggering with IEMI

Intuition. By intentionally injecting IEMI signals into the microswitch circuit to induce a voltage V_{iemi}^{switch} exceeding V_{th} at the NO terminal, fake clicks can be triggered without physically pressing the microswitch.



(a) Previous image. (b) Next image.

Fig. 5. Two adjacent images captured by the image sensor during IEMI injection.

The wire of the printed circuit board (PCB) inside a mouse, serving as a conductor, can function as an antenna. When an attacker emits IEMI signals towards the microswitch circuit, the wire's antenna-like phenomenon enables IEMI signals to electromagnetically couple into the microswitch circuit. Assuming that the power of the transmitted IEMI signal is P_{TX} and the power of the injected IEMI signal in the mouse is P_{RX}. From the Friis transmission equation [41], we have:

$$P_{RX} = G_t G_r (\frac{\lambda}{4\pi D})^2 P_{TX}. \tag{3}$$

G_t and G_r are the gains of the transmitting and receiving antennas, respectively. λ is the wavelength of the IEMI signal and D is the attack distance between the adversary's antenna and the victim mouse. Based on Faraday's law of induction [24], the microswitch circuit generates an induced current after receiving the IEMI signal. This is because when the IEMI signal passes through the microswitch circuit, changes in the internal electric and magnetic fields cause movements of charges in the circuit, resulting in an induced current I_{iemi}^{switch}:

$$I_{iemi}^{switch} = \sqrt{\frac{P_{RX}}{R_c}}, \tag{4}$$

where R_c is the equivalent resistance of the mouse's circuit. In this case, even if the microswitch is not pressed, there still exists an induced voltage V_{iemi}^{switch} at the NO terminal of the microswitch. The induced voltage can be expressed as:

$$V_{iemi}^{switch} = I_{iemi}^{switch} R_{micro}, \tag{5}$$

where R_{micro} represents the equivalent resistance of the microswitch's circuit. According to Eq. 2, a click can be triggered without physically pressing the microswitch as long as the induced voltage V_{iemi}^{switch} is greater than V_{th}, i.e., $V_{iemi}^{switch} > V_{th}$.

4.2 Fake Movement Triggering with IEMI

Intuition. By injecting IEMI signals into the image sensor to create ghost differences between adjacent images, fake movements can be triggered without physically moving the mouse.

According to Sec. 2.2, the image sensor continuously captures images of the surface beneath the mouse, which undergoes a conversion process of "light→charge→voltage→pixel'. However, the image sensor is vulnerable to electromagnetic interference, especially when voltage signals are transmitted from the measurement units to the ADC. There are two reasons for this susceptibility: 1) The long pathway from measurement units to the ADC provides a large area for electromagnetic coupling, which enhances the IEMI signal injection. 2) The internal

Table 1. Results of injecting fake click events on 5 mice.

Models	Triggered Clicks		Triggered Movements
	Left	Right	
HP M10	✓	✓	✓
AOC MS100	✓	✓	✓
ASUS UX300 PRO	✓	✓	✓
Philips 7214	✓	✓	✓
Dell MS116	✓	✓	✓

amplifier can amplify the voltage signal induced by the IEMI signal. In this process, malicious IEMI signals couple to the circuit between the measurement units and the ADC, inducing an additional voltage signal V_{iemi}^{image}, which accumulates to the original voltage signal V_0^{image}. Consequently, the voltage signal input to the ADC, namely V^{image}, is changed to:

$$V^{image} = V_0^{image} + V_{iemi}^{image}. \tag{6}$$

Then, the ADC quantizes V^{image} to generate a image I. Since the ADC sequentially converts the voltages generated by the measurement units into pixel values, injected IEMI signals can affect the image I at the pixel level. Thus, the image I can be expressed as:

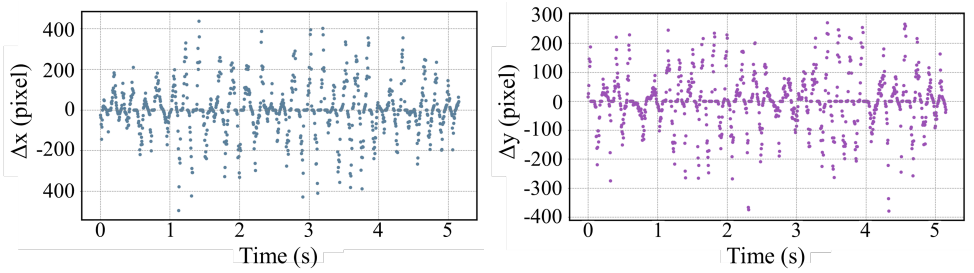
$$\begin{aligned} I &= \frac{V^{image} - V_{min}}{V_{max} - V_{min}} * N \\ &= I_0 + \underbrace{\frac{V_{iemi}^{image}}{V_{max} - V_{min}} * N}_{I_{iemi}}, \end{aligned} \tag{7}$$

where I_0 and I_{iemi} represent the benign image and the IEMI-induced ghost pattern, respectively. V_{min} and V_{max} represent the voltages corresponding to the minimum and maximum pixel values, respectively. N represents the total number of gray levels. To intuitively illustrate the impact of IEMI signals on the image, we visualize the images captured by the image sensor in the Logitech G402 mouse through reverse engineering [30]. Figure 4(a) shows the benign image captured by the image sensor without IEMI signal injection. Yet, when IEMI signals are injected into the image sensor, we can observe that a ghost pattern is superimposed onto the benign image, as shown in Fig. 4(b).

Through IEMI signal injection, attackers can superimpose ghost patterns onto different positions in successive images to introduce differences between adjacent images. Since the mouse's DSP uses the template matching algorithm to detect mouse movements based on template features, the induced ghost patterns must meet two criteria to trigger fake movements: 1) Part of the ghost pattern needs to be superimposed onto the template region of the image. 2) They need to introduce sufficient features into the image templates. Subsequently, the DSP analyzes the positional changes of the ghost patterns between adjacent images, triggering fake mouse movement. For instance, Fig. 5(a) and Fig. 5(b) are adjacent images captured during IEMI signal injection. We can observe that the ghost pattern in Fig. 5(b) is shifted downwards with respect to Fig. 5(a). In this case, a downward mouse movement can be detected by the DSP without any physical manipulation of the mouse.

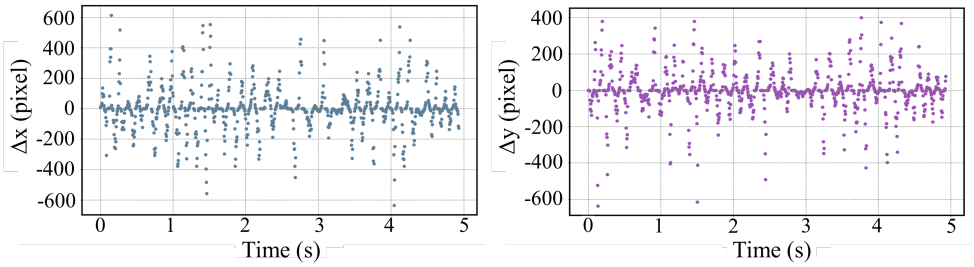
4.3 Preliminary Experiments and Observations

We conduct preliminary experiments to further prove the correctness of the theoretical analysis and the practicality of fake clicks and movements with IEMI injection in the real world.



(a) Variations on the movement direction and speed along x-axis over time for Mouse HP M10.

(b) Variations on the movement direction and speed along y-axis over time for Mouse HP M10.



(c) Variations on the movement direction and speed along x-axis over time for Mouse AOC MS100.

(d) Variations on the movement direction and speed along y-axis over time for Mouse AOC MS100.

Fig. 6. When IEMI signals are injected into two mice (Mouse HP M10 and Mouse AOC MS100), the direction and speed of mouse movement along the x- and y-axes on the screen exhibited random fluctuations over time.

■ **Experiment I: IEMI-Induced Clicks:**

Setup: The overall setup includes a USRP B210 [7], an omnidirectional antenna with 35 dBi gain [4], and a commercial RF power amplifier [5]. The USRP B210 is used to generate adversarial IEMI signals, which are then amplified by the RF power amplifier and finally emitted by the antenna. In this experiment, we test with 5 mainstream mouse models. As the vulnerable IEMI frequency is unclear, we use a 5 MHz step to sweep from 70 MHz to 600 MHz and emit the corresponding IEMI signals. For each frequency, we conduct fifteen 10-second experiments, with the injection power set to 3 W.

Results: As shown in Table 1, it is evident that IEMI signals can induce clicks across various mouse models (5 out of 5), indicating the widespread susceptibility of mice to IEMI attacks. Furthermore, upon successful attacks, we have two observations: 1) **Unpredictable click.** Both the left and right microswitches of the mouse are vulnerable to IEMI attacks, suggesting that IEMI injection can trigger both fake left and right clicks[4] However, for each attack, the triggering of left and right clicks is random. We attribute this randomness to the IEMI-induced voltages at the NO terminals of these two microswitches, as any of them may exceed the threshold voltage V_{th}. 2) **Movement concurrence.** These mice sometimes exhibit mouse movements simultaneously with click events. We think that movement occurs because the IEMI signal also creates detectable differences between adjacent images captured by the image sensor.

■ **Experiment II: IEMI-Induced Movements:**

[4]The middle microswitch on a mouse is typically infrequently used and primarily for scrolling pages—a function that can be replaced by mouse movement. Therefore, this paper only considers the effects of IEMI signals on the left and right microswitches.

Setup: We utilize the same equipment as Experiment I to generate and emit IEMI signals. As mentioned in Sec. 4.2, the key to inducing fake movements in mice with IEMI signals lies in creating differences between adjacent images. Thus, in this experiment, we modulate a random signal onto carrier signals at different frequencies. We sweep the carrier frequencies from 70 MHz to 600 MHz with a 5 MHz step and emit corresponding IEMI signals until observing mouse movements. For each frequency, we perform fifteen 5-second trials with the injected power set to 3 W.

Results: All the mice (5 out of 5) are susceptible to IEMI-triggered movements, which underscores the pervasive vulnerability to such an attack. Figure.6 illustrates the variation in the direction and speed of mouse movement along the x-axis and y-axis over time on the screen when IEMI signals are injected into two mice (i.e., Mouse HP M10 and Mouse AOC MS100). We can observe two phenomena: 1) **Randomness in movements.** The movements of these mice appear to be random, in terms of both the direction and speed. This randomness suggests that these mice do not move in a specific direction or speed during the attack. We infer that this unpredictability is due to the injected random signal, which induces stochastic differences in the adjacent captured images; 2) **Click occurrence.** Some mice induce click events simultaneously with movement events. We consider that clicks occur because the IEMI signals also induce voltage that exceeds the threshold V_{th} at the NO terminal of the microswitch.

■ **Challenge**: Our preliminary experiments have demonstrated the susceptibility of various mouse models to IEMI. Consequently, it is feasible for an attacker to inject fake clicks and movements into the mice and further take charge of the computers connected to them without any physical contact. However, achieving precise control over mouse clicks and movements by IEMI injection needs to address several challenges: 1) Random triggering of left and right clicks; 2) Unpredictability of mouse movement in terms of direction and speed; 3) The possibility of simultaneous occurrence of click and movement events, leading to mutual interference. As a next step, we will explore how to make fine designs of the IEMI signals to achieve two-side clicks as well as direction- and speed-controllable movements.

5 PUPPETMOUSE DESIGN

We present PuppetMouse, the first attack that can deliberately induce click and movement events on the mouse via IEMI injection. To ensure that the IEMI signals can precisely manipulate the mouse, it is crucial to design PuppetMouse in three aspects: 1) precise control over two-side (i.e., left and right) clicks; 2) direction- and speed-adjustable movements; and 3) eliminating interference between click and movement.

Figure 7 outlines the workflow of PuppetMouse. Upon the theoretical analysis in Sec.4, we first design the IEMI signals injected into the microswitch circuit and the image sensor to enable controllable clicks and movements, respectively. Considering that the same IEMI signal may be effective for both of the components, and triggers click and movement events simultaneously, we further optimize the IEMI signals based on the inherent operational differences of these two components. The optimization allows these two events to be triggered independently of each other.

5.1 Controllable Two-side Click

The key to generating fake clicks via IEMI is to induce a voltage (i.e., V_{iemi}^{switch}) on the NO terminal of the microswitch that exceeds V_{th}. That is to say, the effectiveness of triggering fake clicks depends on V_{iemi}^{switch}. To effectively trigger two-side clicks, we first explore the relationship between induced voltage V_{iemi}^{switch} and IEMI signal characteristics (e.g., the frequency and amplitude). Then, we investigate how to adjust these characteristics to achieve precise control of click sides.

The microswitch circuits are likely to couple with high-frequency IEMI signals as they lack filters [26]. Consequently, we design an IEMI signal consisting of two parts: a high-frequency carrier signal and a modulation

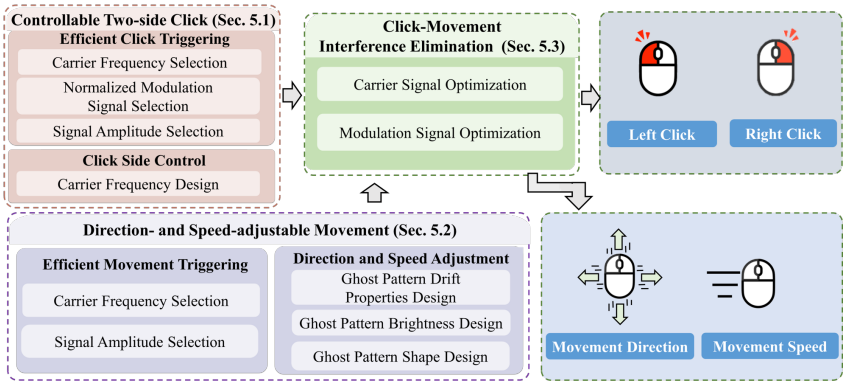


Fig. 7. Workflow of PuppetMouse attack.

signal. The former serves as a data transmission medium, akin to a channel between the microswitch circuit and the transmitting antenna. The latter carries the 'command' that guides the click generation. We represent the emitted IEMI signal as $s(t) = m(t) * Acos(2\pi f_c t)$, where $m(t)$ is the normalized modulation signal, f_c is the carrier frequency, and A is the signal amplitude. Except for these parameters, the duration of the emitted IEMI signal also needs to be considered. This is because the debounce mechanism in the microswitch circuit identifies excessively short switch signals as noise [16]. Hence, to induce effective clicks, the duration of the IEMI signal must exceed a certain threshold, T_{switch}. At this point, the transmitting power of the IEMI signal can be expressed as:

$$P_{TX}(t) = \lim_{T\to\infty} \frac{1}{T} \int_0^T |s(t)|^2 dt, \quad T > T_{switch}. \tag{8}$$

Combing Eq. 3 and 8, the power of the received IEMI signal can be expressed as:

$$\begin{aligned} P_{RX} &= G_t(f_c) * G_r(f_c) * (\frac{\lambda(f_c)}{4\pi D})^2 * P_{TX} \\ &= G_t(f_c) * G_r(f_c) * (\frac{\lambda(f_c)}{4\pi D})^2 * \frac{A^2}{2} * P_m, \end{aligned} \tag{9}$$

where $G_t(f_c)$ and $G_r(f_c)$ represent the gains of the transmitting antenna and microswitch circuit at different carrier frequencies, respectively. $\lambda(f_c)$ is the wavelength of the IEMI signal at different carrier frequencies. P_m is the power of the normalized modulation signal. From this equation, we can find that P_{RX} is primarily determined by the carrier frequency (f_c), the normalized modulation signal ($m(t)$), and the signal amplitude (A). Additionally, combing Eq. 4 and 5, we can derive that the induced voltage V_{iemi}^{switch} is proportional to P_{RX}. Therefore, V_{iemi}^{switch} also depends on these parameters (i.e., f_c, $m(t)$, and A). In the following, we introduce how to design them to induce sufficiently high voltage and trigger effective clicks.

Carrier frequency f_c: According to [53], the range of carrier frequencies effectively coupled onto the circuit depends on the length of the circuit. To optimize signal coupling, the carrier frequency must exceed $c/50L$, where c and L represent the speed of the light and the length of the circuit, respectively. Building on this foundation, we adopt sweeping frequency technology to identify the carrier frequencies that can trigger fake clicks even at low IEMI signal power. These frequencies are considered as vulnerable frequencies.

Normalized modulation signal $m(t)$**:** Since the intensity of the induced voltage is proportional to the power of the modulation signal (P_m), we choose a direct current (DC) signal as the modulation signal due to its high power.

Signal amplitude (A): To execute an IEMI attack over extended distances, we employ a power amplifier to enhance the signal amplitude. This can effectively increase the attack distance, improving the effectiveness and practicality of PuppetMouse.

With the above designs, we can effectively trigger clicks. However, since the microswitches of the left and right clicks work on the same principle, a modulation signal may trigger the left and right clicks concurrently. To address this issue, we further fine-tune the IEMI characteristics. Specifically, as the main difference between the left and right microswitches lies in the length of their circuits (L), their vulnerable frequency ranges (f_c) are also different. Hence, to specify the click side, we select the non-overlapping vulnerable frequencies specific to the left or right microswitch circuits and separately apply them to launch desired click attacks.

5.2 Direction- and Speed-adjustable Movement

Similar to IEMI-induced click generation, the IEMI signal that manipulates movement is also composed of a carrier signal and a modulation signal, which can be expressed as $s(t) = A * m(t) * cos(2\pi f_c t)$. For the carrier signal, we regulate its carrier frequency and amplitude similar to the click-triggered IEMI signal. As for the normalized modulation signal $m(t)$, since it contains different 'commands' to guide the fake movement triggering, we perform tailored designs to achieve direction and speed control.

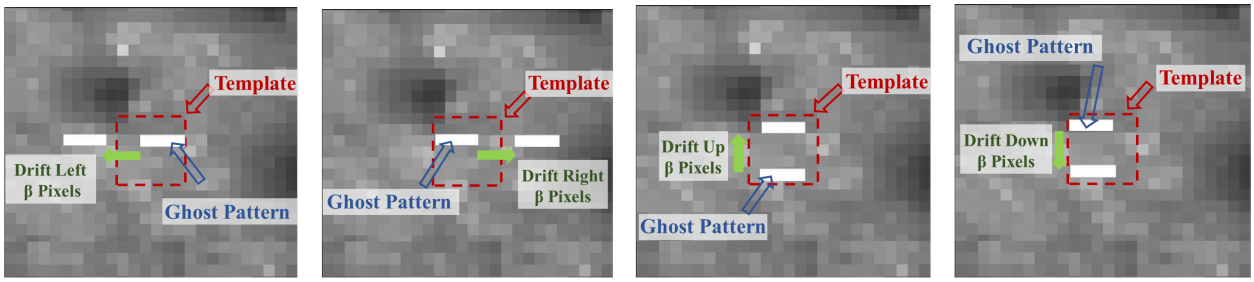
To be specific, according to the theoretical analysis in Sec. 4.2, the key to triggering fake mouse movements via IEMI is to continuously inject ghost patterns into captured images and create differences between them. Thus, to achieve direction- and speed-adjustable mouse movement, we first need to craft a delicate ghost pattern, and then inject it into each captured frame and move it at the desired direction and speed along successive frames. To make the attack precise and robust, we focus on four properties of the ghost pattern, including drift direction, drift speed, brightness, and shape. According to the principle of the template matching algorithm, the ghost pattern drift characteristics (i.e., direction and speed) help the template matching algorithm infer the movement direction and speed of the mouse. Meanwhile, the brightness and shape of the ghost pattern affect the accuracy of template matching, and thus determine the robustness of the attack. To accurately control these four properties with the normalized modulation signal $m(t)$, we first analyze the relationship between them. In particular, $m(t)$ can be expressed as:

$$m(t) = W[2\pi f_m t] * \sum_{n=0}^{\infty} rect(\frac{t - n\frac{1}{f_m}}{T_1}). \tag{10}$$

It can be observed that $m(t)$ has three configurable parameters: the modulation frequency f_m, the duration of IEMI injection in each cycle T_1, and the waveform W. By adjusting these parameters, we can tailor the ghost pattern. 1) f_m determines the injection frequency of the ghost pattern. By adjusting f_m, we are able to vary the drift direction and speed of the ghost pattern between consecutive frames. 2) T_1 determines the duration of IEMI injection in each cycle. By adjusting it, we can control the shape of the ghost pattern. 3) W determines the IEMI signal strength. Its variation will affect the brightness of the ghost pattern. In the following, we demonstrate how to model these parameters (i.e., f_m, T_1 and W) to manipulate the drift properties (i.e., direction and speed), brightness, and shape of the ghost pattern.

■ **Drift Property of the Ghost Pattern.** The essence of causing ghost pattern drift is to generate the difference between the image sensor's frame rate f_0 and the modulation frequency f_m. Thus, we need to explore the relationship between the drift properties (i.e., direction and speed) of the ghost pattern and the frequencies (i.e., f_0 and f_m). An image captured by the image sensor consists of m rows and m columns. We denote the benign image [5] without IEMI pollution as a pixel sequence $I_0(n)$, $n \in (1, m * m)$. When an IEMI signal is injected into the image sensor, a ghost pattern will be superimposed onto the benign image. Assuming that the

[5]The benign image remains unchanged when the mouse is static.



(a) Left. (b) Right. (c) Up. (d) Down.

Fig. 8. The ghost pattern drifts β pixels in four directions: left, right, up, and down.

ghost pattern has i pixels, and the ghost pattern shifts x pixels forward along adjacent images. I_1 and I_2 are two adjacent images during IEMI injection, that is, $I_1 = [I_0(1:k), P + I_0(k+1:k+i), I_0(k+i+1:m*m)]$ and $I_2 = [I_0(1:k-x), P + I_0(k-x+1:k-x+i), I_0(k-x+i+1:m*m)]$. P denotes the ghost pattern. As P is more evident than the benign image I_0, P can overlay I_0. Therefore, I_1 and I_2 can be rewritten as $I_1 = [I_0(1:k), P, I_0(k+i+1:m*m)]$ and $I_2 = [I_0(1:k-x), P, I_0(k-x+i+1:m*m)]$. At this time, the injection frequency f_m of the ghost pattern can be expressed as :

$$f_m = \frac{m*m}{m*m-x} * f_0. \tag{11}$$

According to the above analysis, we can manipulate the movement direction and speed of the mouse by the following means:

i) Direction: To make the mouse move to a specific direction by β pixels, we need to:

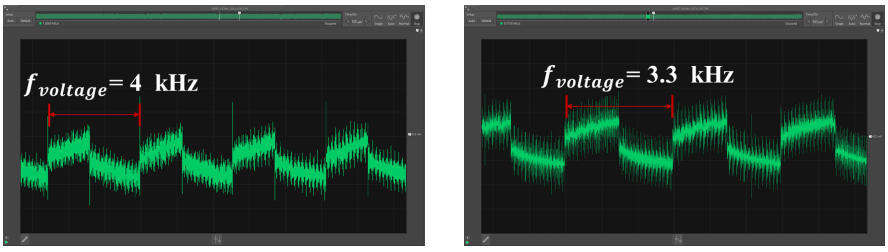
(1) Left: set f_m to $\frac{m*m}{m*m-\beta} * f_0$. As shown in Fig.8(a), where $x = \beta$.
(2) Right: set f_m to $\frac{m*m}{m*m+\beta} * f_0$. As shown in Fig.8(b), where $x = -\beta$.
(3) Up: set f_m to $\frac{m*m}{m*m-m*\beta} * f_0$. As shown in Fig.8(c), where $x = m * \beta$.
(4) Down: set f_m to $\frac{m*m}{m*m+m*\beta} * f_0$. As shown in Fig.8(d), where $x = -m * \beta$.

However, in real scenarios, we face a problem: not all image sensor manufacturers publish the frame rate (i.e., f_0) of their devices. To address this challenge, we utilize the voltage signal of the image sensor to infer its frame rate. This is because the voltage signal of the image sensor can reflect its internal periodic operational condition. Each image capture can be considered as one cycle. Therefore, the frequency of the voltage signal equals the frame rate of the image sensor. To demonstrate the feasibility of this method, we select two mice (HP M10 and Dell MS116) whose frame rates have been published to conduct a validation experiment. Their frame rates are 4 kHz and 3.3 kHz, respectively. Then, we measure the voltage signals of these two mice with an oscilloscope when they are in movement. As shown in Fig. 9, the frequencies of voltage signals $f_{voltage}$ of mice HP M10 (i.e., 4 kHz) and Dell MS116 (i.e., 3.3 kHz) are consistent with their frame rates, which proves the feasibility of this method.

ii) Speed: The movement speed of the mouse is proportional to the number of pixels that the ghost pattern drifts between adjacent images (i.e., β). Thus, we can control the speed of the mouse by adjusting β [6].

■ **Brightness of the Ghost Pattern.** To make the attack robust, the brightness of the ghost pattern should be sufficiently high to ensure that the ghost pattern is visible and stable when superimposed on a benign image. Particularly, the placement manner of the mouse is variable, e.g., the mouse may be placed on different pads,

[6]It is worth noting that the ghost pattern drift of each frame cannot exceed the length of the image (i.e.,m).



(a) Mouse HP M10. (b) Mouse Dell MS116.

Fig. 9. The voltage signal frequencies of two mice.

rendering the captured benign image uncertain. Besides, it is worth noting that the IEMI can only increase the pixel values (i.e., the brightness), not decrease them. In this case, injecting a white ghost pattern emerges as the optimal solution, given that the pixel value corresponding to white is the largest. As long as the IEMI-induced voltage in the image sensor exceeds the voltage V_{max} (i.e., the voltage corresponding to the maximum pixel value), the ghost pattern will appear white regardless of the benign image. To ensure a white ghost pattern, we need to inject a sufficiently strong IEMI-induced voltage into the image sensor. According to Sec.5.1, the strength of the induced voltage depends on the power received by the image sensor. Therefore, we choose a DC voltage signal as the modulation signal due to its high power characteristics, where the waveform W satisfies $W = DC$.

■ **Shape of the Ghost Pattern.** Recall that the image sensor employs the template matching algorithm to detect the mouse movement. When the ghost pattern moves into the template region of the captured image, the mouse detects differences between adjacent images based on the ghost pattern to determine its movement. Therefore, when designing the shape of the ghost pattern, we need to consider the following two aspects: 1) The ghost pattern should not be too small, or it cannot introduce sufficient features to ensure accurate template recognition. 2) The size of the ghost pattern should not exceed that of the template; otherwise, the mouse may detect multiple regions on the image, misleading the template matching.

To meet the above requirements, the shape of the ghost pattern should be $1 * H$, where H represents the length of the template. With such a shape, the ghost pattern not only introduces enough features into the template but also avoids misjudgment of the mouse movement caused by template matching. To generate a qualified ghost pattern, the duration of IEMI injection in each cycle T_1 should be set to:

$$T_1 = \frac{H}{m * m * f_m} \tag{12}$$

However, in practical scenarios, the image sensor manufacturers do not publish the shape of the template. Hence, we cannot directly determine the shape of the ghost pattern. To address this issue, we propose an adaptive algorithm to calculate the optimal shape of the ghost pattern. The insight of the algorithm is that only when the shape of the ghost pattern is $1 * H$, the cursor on the computer screen can accurately move according to the drift of the ghost pattern. So, we estimate the optimal shape of the ghost pattern by varying the shape of the ghost pattern and observing the resulting attack effectiveness. Specifically, we establish four reference directions (i.e., up, down, left, and right) and calculate the modulation frequency f_m to match these four reference directions, respectively. In each movement, we set the initial position of the cursor at the center of the screen. For each direction, we continuously collect the positions of the cursor on the screen at intervals of Δt for D times. Next, we calculate the angle α_i between the vector from the initial position to each cursor position i ($i \in (1, D)$) and the reference direction. If the average angle $\overline{\alpha}$ is less than a pre-set threshold γ, we consider that the mouse moves towards the reference direction. To find the optimal shape of the ghost pattern, we gradually adjust the duration

of IEMI injection in each cycle T_1 and observe the movement of the cursor. Once the cursor can accurately move in all four reference directions, the adopted ghost pattern shape is considered optimal.

5.3 Click-Movement Interference Elimination

The high integration of circuits in the mouse may cause the IEMI to couple onto the click circuit and the image sensor simultaneously. As a result, an attacker trying to generate fake clicks may unintentionally trigger fake movements, or vice versa. To eliminate the click-movement interference, we further optimize the IEMI signal (including carrier and modulation signals) to guarantee separate event triggering.

■ **Carrier Signal Optimization.** As introduced in Section 5.1 and Sec.5.2, we determine the ranges of the vulnerable frequencies for both the click circuit and the image sensor, respectively. However, the vulnerable frequency ranges of click circuits and image sensors may partially overlap, resulting in click-movement interference. To solve this problem, we select non-overlapping carrier frequencies for the click circuit and image sensor, respectively. In this way, the attacker can minimize the impact of the carrier signal on the components corresponding to the movement when inducing a fake click, or vice versa.

■ **Modulation Signal Optimization.** The modulation signal carries the 'command' that guides mouse clicks or movements. Nevertheless, as the principle of evoking click and movement is similar, i.e., introducing a voltage signal in the corresponding electronic component, the modulation signal designed above may concurrently cause fake click and movement. To suppress the interference induced by such 'generality', we first analyze the working principles of the click circuit and image sensor to figure out if it is the signal designed for clicks that are affecting movements, or if it is the other way around. Then, based on the analysis, we optimize the modulation signals corresponding to these two events to ensure the Independence of each attack.

1) May click-targeted modulation signal induce movement? The core of IEMI-triggered movement is that the ghost patterns injected by IEMI create detectable differences between adjacent images. Due to the high integration of the mouse circuit, the click-targeted modulation signal may induce a voltage in the image sensor and superimpose ghost patterns onto the benign images. The modulation signal that triggers the click is a DC signal with a duration t that greater than T_{switch}. Typically, T_{switch} takes several milliseconds [6, 44]. Without any user interaction, the mouse remains in an idle state. At this point, the image sensor captures images at intervals of approximately ten milliseconds (denoted as T_{idle}) [7]. If the duration of the IEMI signals exceeds T_{idle}, it may cause the ghost pattern to appear in multiple frames, resulting in differences between them. These differences may be detected by the template matching algorithms, and further lead to mouse movement. To prevent mouse movement when IEMI triggers a click, the modulation signal should not introduce continuous, identifiable changes across images. To this end, the duration t for the click-targeted modulation signal needs to be less than T_{idle}. This ensures that the IEMI signal is injected into at most two images. Even if the image sensor detects differences between two adjacent images, the mouse will not move. Therefore, the duration t of the IEMI signal for triggering an individual click should satisfy $T_{switch} < t < T_{idle}$.

2) May movement-targeted modulation signal induce click? The modulation signal that triggers movements is a rectangular pulse signal with a frequency of f_m. The injection time T_1 (typically tens microseconds) of the IEMI signal in each cycle is shorter than the duration T_{switch} required to trigger a click (typically several milliseconds). Thus, the movement-triggered modulation signals would not cause mouse clicks.

6 PUPPETMOUSE EVALUATION

This section first introduces the implementation of PuppetMouse and then evaluates its effectiveness.

[7]The mouse manufactures usually do not provide the parameter T_{idle}. We read T_{idle} from the oscilloscope through voltage signals.

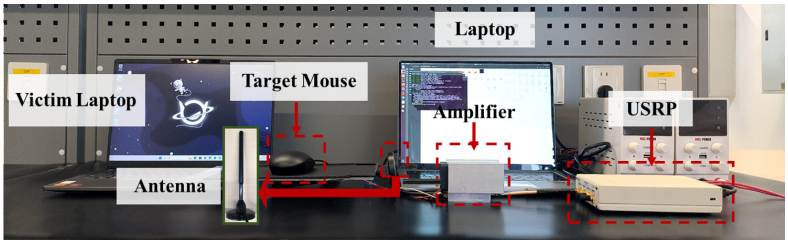


Fig. 10. The setup of PuppetMouse attack.

6.1 Setup

The IEMI injection setup comprises a software-defined radio USRP B210 [7], an omnidirectional antenna with 35 dBi gain [5], and a commercially available RF power amplifier [4], as shown in Fig 10. The USRP B210 is responsible for generating adversarial IEMI signals, which are subsequently amplified by the RF power amplifier and transmitted through the antenna placed beneath the target mouse. The maximum injection power is set to 4 W. As shown in Table. 2, we evaluate our PuppetMouse attack on 14 different models of mice from 11 popular brands. Among them, 11 are wired mice and three are wireless ones. Unless specially, the mice are connected to a Lenovo Xiaoxin Pro14 laptop (victim computer) for evaluation. We also use another laptop with Intel(R) Core(TM) i5-8265U CPU to craft and generate IEMI signals.

6.2 Metrics

To evaluate the effectiveness of PuppetMouse, we define specific metrics for IEMI-induced clicks and movements, respectively.

■ **Click.** We focus on three key metrics: click success rate, response delay, and interference rate. The click success rate denotes the proportion of attacks that successfully trigger the desired click event without movement. The response delay is the time it takes to trigger the successful click event since the attack starts. The interference rate refers to the proportion of attacks that mistakenly generate movements or undesired side clicks at the same time.

■ **Movement.** For fake movement assessment, we define five metrics: angular deviation, movement speed, response delay, interference rate, and response rate. The angular deviation is the angle between the movement trajectory (i.e., the vector from the movement starting point to the endpoint) and the target direction. The movement speed refers to the number of pixels the cursor moves on the screen per second in the target direction. The response delay is the time it takes to inject the movement event since the attack starts. The interference rate refers to the proportion of attacks that mistakenly induce clicks at the same time. The response rate refers to the proportion of attacks that indeed trigger movements.

6.3 Overall Effectiveness

We first evaluate the performance of triggering (1) two-side clicks and (2) direction-and speed-adjustable movements via IEMI signals on these 14 mice. In each experiment, the antenna is placed directly under the center of the victim mouse. The injection power is set to 4 W.

6.3.1 Click. In this experiment, we first use frequency sweeping for each mouse to identify the vulnerable frequencies of its left and right microswitch circuits, respectively. Next, we determine the minimum time required to trigger a click. Building upon the above steps, we inject the corresponding IEMI signal into the left or right microswitch circuit using the IEMI injection setup, respectively. For each microswitch circuit, we launch attack

Table 2. The attack results of injecting two-side click events on 14 mice.

Mouse Index	Model	Left Click			Right Click		
		CSR (%)	RD (ms)	IR (%)	CSR (%)	RD (ms)	IR (%)
1	HP M10	95	2.7	2	94	2.4	0
2	AOC MS100	88.5	2.9	0	87.5	2.9	0
3	ASUS UX300 PRO	97.5	2.5	0	95.5	2.8	2.5
4	Philips 7214	70.5	2.8	1.5	76	2.6	2
5	Dell MS116	79	2.7	4	72.5	2.7	3
6	Philips M101	93.5	2.1	3	95.5	2.9	0
7	Logitech M90	17	2.9	0	20.5	3.1	0
8	Komen Wireless M20	81	2.5	0	85.5	2.5	0
9	Philips Wireless M211	58.5	2.5	2	54	2.8	3.5
10	Acer M115	91	2.3	2	89.5	2.6	0
11	Aigo Q21	98.5	2.5	0	97.5	2.6	0
12	Acer Wireless OMR060	33.5	2.7	0	27	2.9	0
13	Lenovo M120	25.5	2.7	4.5	21	2.6	2.5
14	BOW M136U	23.5	2.9	0	16	3	0

trials 200 times and measure the click success rate (CSR), response delay (RD), and interference rate (IR). The summary of detailed attack results is reported in Table. 2.

From Table. 2, it can be observed that all 14 mice are susceptible to click- triggered attacks via IEMI signals, which indicates the prevalence of this IEMI vulnerability. Meanwhile, most of the mice (9 out of 14) can achieve click success rates over 70%, which confirms the effectiveness of this two-side IEMI attack. Nevertheless, a few mice have click success rates below 40%. We deduce that this may be caused by their thicker PCBs [40] or shielding plastic layers over the PCBs [14], which prevent them from IEMI signals. Notably, it can be seen that the mice's interference rates are all below 5%, demonstrating the effectiveness of our click-movement interference elimination method. Moreover, the average response delay of these 14 mice is less than 3 ms. Such a short time validates that our attack can be realized in real time.

Furthermore, we observe that different mouse models exhibit varying performance in terms of click success rate, interference rate, and response delay when subjected to IEMI-induced two-side clicks. These differences are primarily due to internal hardware variations in mice. Specifically, (1) Click success rate. This depends on the efficiency of IEMI signal coupling into the mouse's micro-switch circuit. Factors such as the thickness and material of the mouse shell, as well as the thickness and shielding layers of the PCB would affect the coupling efficiency of the IEMI signal. Thicker shells with higher conductivity and magnetic permeability shield against electromagnetic interference more effectively, resulting in a lower click success rate. Additionally, PCB shielding layers and filter circuits can also reduce click success rates. (2) Interference Rate. This depends on whether the IEMI signal can accurately couple to the target microswitch circuit. According to Section 5.1, controllable two-side clicks are achieved by adjusting the carrier frequencies of the IEMI signals, which are determined by the length of the left and right micro-switch circuits. In this case, the microswitch circuits on both sides with similar lengths correspond to overlapping susceptible frequency ranges, which results in high interference rates. Conversely, when the difference in circuit length is significant, the interference rate is low. (3) Response delay. This is determined by the response speed of the mouse's internal processor, buffering mechanism, and protection circuits. Higher response speeds lead to shorter response delays, while lower speeds have the opposite effect.

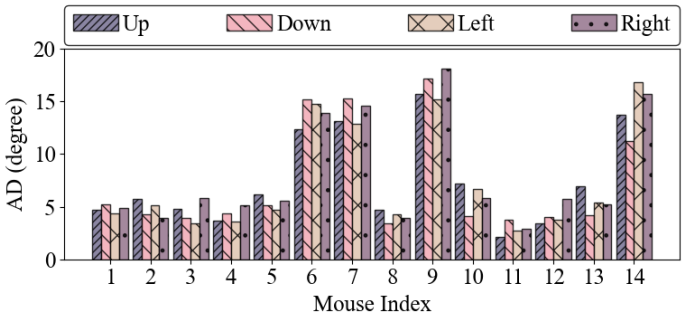
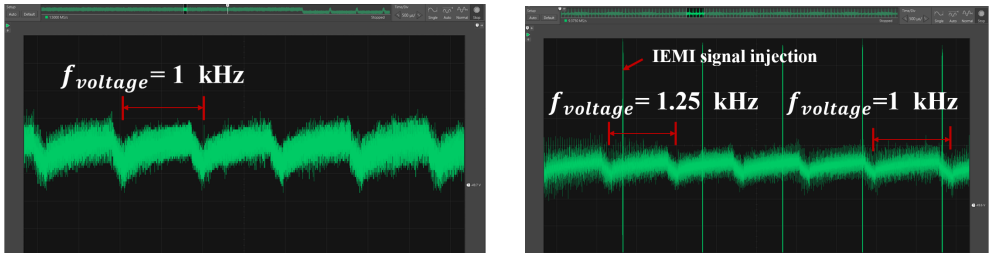


Fig. 11. The angular deviation (AD) of 14 mice in the four target directions (i.e., up, down, left, right)



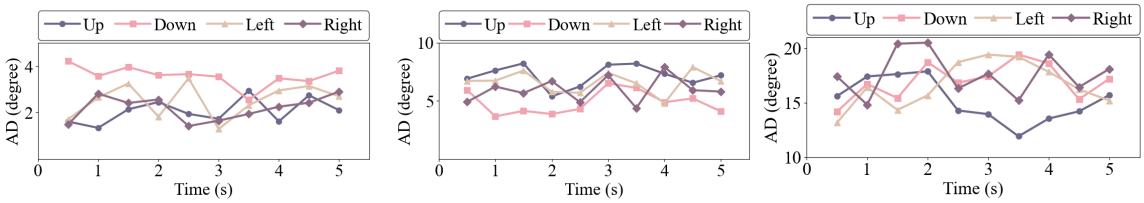
(a) Without IEMI injection.

(b) During IEMI injection.

Fig. 12. The voltage frequency of Mouse BOW M136U.

6.3.2 Movement. In this experiment, we evaluate the performance of IEMI-triggered movement in terms of direction and speed, respectively.

■ **Direction.** For each mouse, we first determine the optimal duration of IEMI injection in each cycle based on the method in Sec.5.2. To make the mouse appear anywhere on the screen, we only need to consider four target directions: up, down, left, and right. For each target direction, we continuously adjust the modulation frequency of the IEMI signal to ensure that the ghost pattern drifts by one pixel toward the target direction on each frame. The default initial position of the mouse is located at the center of the screen. For each target direction, we launch attack trials 100 times. Each IEMI attack lasts 5 seconds. The resulting angular deviations (AD) are shown in Fig. 11. It can be found that most mice (10 out of 14) have small deflection angles (within 7 degrees) in all four target directions. Only a minority of mice deflect more than 10 degrees in each target direction. To further investigate the reasons for these large deviation angles, we use an oscilloscope to observe the changes in voltage signals of these mice with large angular deviation during IEMI injection. As shown in Fig. 12, the voltage signal frequencies $f_{voltage}$ (i.e., the frame rate of the image sensor) of these mice (e.g., Mouse BOW M136U) become unstable during IEMI injection. We believe that this may be due to the interference of IEMI with the mouse's switch circuit. Such unstable frame rates ultimately result in significant angular deviations of these mice. Furthermore, we observe that the interference rate for most mice is 0. Even the highest interference rate is only 2%, which indicates the outstanding effectiveness of our click-movement interference elimination method. Additionally, the highest response delay is less than 4 ms and the lowest response rate is larger than 96%, further showcasing the feasibility of such IEMI attacks.



(a) Mouse Aigo Q21. (b) Mouse Acer M115. (c) Mouse Philips Wireless M211.

Fig. 13. The changes in the angular deviations of three mice over time.

During the experiment, we observe differences among various mice in terms of angular deviation, response delay, interference rate, and response rate. These differences are primarily determined by the internal hardware characteristics of the mice. The impact of the hardware on response delay, interference rate, and response rate is detailed in Section 6.3.1. The angular deviation depends on the stability of the mouse's voltage signal frequency (i.e., the frame rate of the image sensor), which is affected by the impact of movement-targeted IEMI signals on the mouse's switch circuitry. On one hand, when the length of the switching circuitry is relatively similar to that of the image sensor, movement-targeted IEMI signals can couple into the circuitry, leading to a large angular deviation. On the other hand, if the switch circuitry includes electromagnetic shielding measures (such as shields or filters), the IEMI signals cause less interference, resulting in a small angle deviation.

Moreover, to provide a fine-grained illustration of the IEMI attacks, we select three mice with the best (Aigo Q21), moderate (Acer M115), and worst (Philips Wireless M211) performance. Specifically, each mouse is tested in four target directions, with each experiment lasting 5 seconds. For each target direction, we measure the angular deviation (AD) of its movement trajectory from the target direction every 0.5 seconds. Figure. 13 shows the changes in the AD of these three mice over time. We can see that the angular deviations of Mouse Aigo Q21 and Mouse Acer M115 remain stable over time (within 3.5 degrees), suggesting a relatively steady movement trajectory. However, Mouse Philips Wireless M211 exhibits significant variations in angular deviation (approximately 6.5 degrees). This significant variation is caused by the instability in the frame rate of the image sensor during IEMI injection.

■ **Speed.** According to Sec.5.2, speed-adjustable mouse movement is achieved by controlling the drift speed of the IEMI-induced ghost pattern. For each mouse, we adjust the modulation frequency of the IEMI signal to make the ghost pattern drift by one, two, three, or four pixels between adjacent images. This results in corresponding mouse movement speed denoted as v_1, v_2, v_3, and v_4. In each experiment, the mouse's initial position is set to the center of the screen. The injection duration of the IEMI signal for each mouse lasts 10 seconds[8]. We conduct such experiments in the four target directions (i.e., up, down, left, and right) and record the results of the movement speed in Table 3. We can observe that in each target direction, the mouse's movement speed is proportional to the drift speed of the ghost pattern. This result validates the feasibility of implementing speed-adjustable movement by controlling the frequency of the IEMI modulation signal. Furthermore, we have observed that despite the ghost pattern drifting at the same speed, different mice demonstrate varying movement speeds. This variance is influenced by the sizes of both the captured images and the templates. Specifically, smaller images and larger templates lead to faster mouse movement speeds, whereas larger images and smaller templates result in slower speeds.

Based on the above experiments, we can utilize crafted IEMI signals to move the mouse to any position on the screen and perform two-side clicks, thereby achieving a remote takeover of the connected computer.

[8]During the attack, if the mouse reaches the edge of the screen, we will reset it to the center of the screen.

Table 3. Movement speed of 14 mice in four target directions.

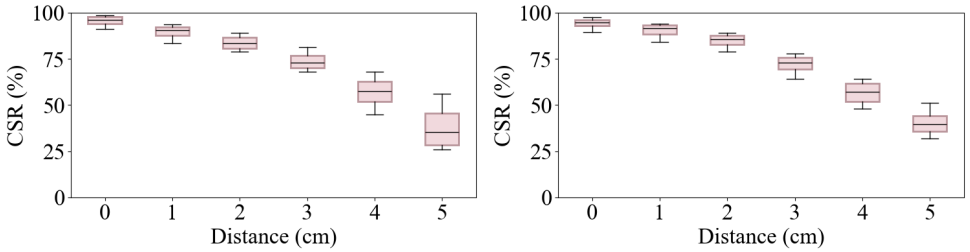
Model	Movement Speed (pixels/s)															
	Up				Down				Left				Right			
	v_1	v_2	v_3	v_4	v_1	v_2	v_3	v_4	v_1	v_2	v_3	v_4	v_1	v_2	v_3	v_4
HP M10	56	95	135	207	52	90	134	189	46	87	123	157	51	97	141	193
AOC MS100	63	109	169	207	61	114	163	214	57	103	151	193	65	121	173	234
ASUS UX300 PRO	31	52	84	113	39	67	114	153	30	54	89	119	27	50	79	97
Philips 7214	53	92	139	191	59	114	172	230	49	91	143	190	52	83	129	141
Dell MS116	25	41	71	82	29	51	79	92	22	39	57	83	24	37	71	92
Philips M101	37	71	104	142	32	61	92	123	29	53	85	116	39	76	117	152
Logitech M90	22	39	61	79	17	27	49	62	25	47	71	92	23	39	64	89
Komen Wireless M20	40	77	111	157	42	78	119	152	38	74	111	147	43	81	124	159
Philips Wireless M211	35	63	99	138	31	57	87	116	41	76	114	156	37	69	105	139
Acer M115	20	31	54	79	24	41	68	89	21	37	59	78	17	31	49	61
Aigo Q21	54	101	154	209	57	107	166	221	51	94	142	191	59	114	173	232
Acer Wireless OMR060	39	73	109	147	31	59	91	117	31	54	87	111	34	61	94	129
Lenovo M120	45	87	123	169	47	83	135	176	43	81	127	168	54	94	143	177
BOW M136U	37	69	101	140	31	58	87	116	31	61	91	117	33	57	94	125

6.4 Effect of Attack Distance

In this part, we explore the impact of the attack distance (i.e., the distance between the attack device and the mouse) on the IEMI-induced clicks and movements. In the experiments, the attack distance is varied from 0 cm to 5 cm in steps of 1 cm. The injection power is set to 4 W.

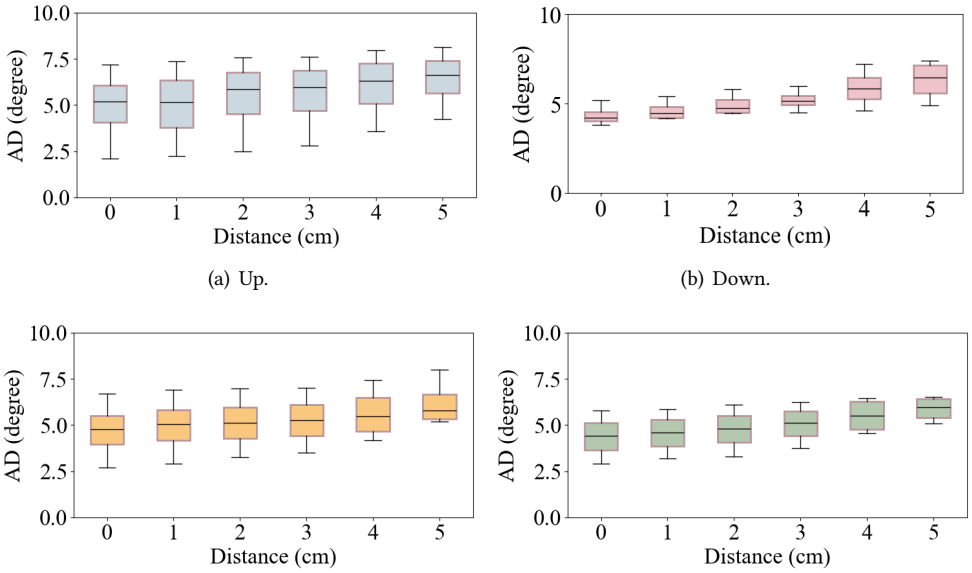
■ **Click.** For each attack distance, we perform 1600 attacks on both the left and right click circuits of the mice and measure the corresponding click success rates (CSR) and the interference rates. The two-side click success rates are shown in Fig. 14. We can find that within a range of 5 cm, IEMI can effectively trigger two-side clicks. It can also be found that the click success rates decrease with the increasing attack distance. This is because the IEMI power injected into the mouse intends to decay with the increasing attack distance. Due to the typical table thickness of 1/2 in (12.7 mm) or 5/8 in (15.9 mm) [17], the induced clicks success rate can surpass 79% in this case. This indicates that IEMI-induced clicks will be effective in real attack scenarios. It is worth noting that a skilled attacker can enhance their attack coverage by utilizing custom directional antennas with increased power supply capabilities. Besides, the interference rates remain below 5% at varying attack distances, demonstrating the robustness of our click-movement interference elimination method.

■ **Movement.** At each attack distance, we attempt to move the mouse in four target directions (i.e., up, down, left, right). To this end, we fine-tune the modulation frequency of the IEMI signal to drift the ghost pattern by one pixel toward the target direction on each frame. For each target direction, we conduct 200 attacks on the mice, with each attack lasting 5 seconds. The default initial position of the mice is at the center of the screen. During each experiment, we measure the angular deviation, the interference rate, and the response rate to quantify the performance of IEMI-triggered movement. Figure. 15 shows the angular deviations (AD) of the mice in the four target directions at different attack distances. Although increasing attack distance results in a slight increase in angular deviation, the change in angular deviation for each mouse remains within 3.4 degrees. Such small angular deviation demonstrates the robustness of IEMI-triggered movement across various attack distances. Moreover, the interference rates remain below 2% and the response rates remain above 97.5% at different attack distances.



(a) The left click circuit. (b) The right click circuit.

Fig. 14. The two-side click success rates (CSR) at different attack distances.



(a) Up. (b) Down.

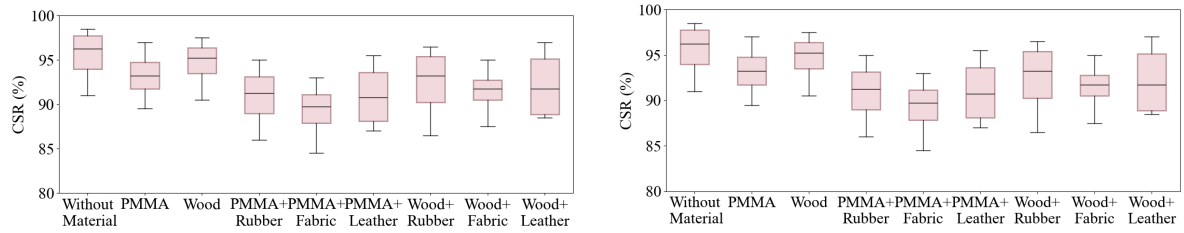
(c) Left. (d) Right.

Fig. 15. The angular deviations (AD) of IEMI-induced movements at different attack distances in four target directions.

This also proves the robustness of our click-movement interference elimination method against attack distance variations. Based on the above results, we can conclude that within an attack distance range of 5cm, the impact of IEMI on the image sensor remains relatively stable and small. Thus, the attacker can still achieve speed- and direction-adjustable mouse movement under large attack distances.

6.5 Effect of Material

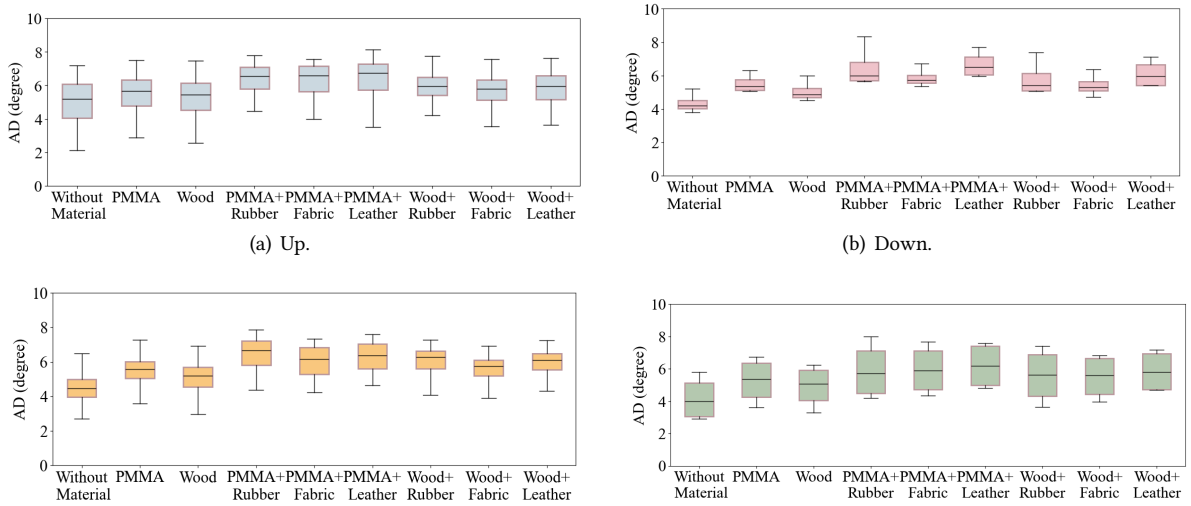
In practice, users may use tables and mouse pads made of different materials, which are between the mouse and the attack device. To further validate the robustness of PuppetMouse, we assess the impacts of 5 mainstream materials, including two popular table materials, i.e., PMMA ($\varepsilon = 2.7 \sim 4.0$) and wood ($\varepsilon = 1.2 \sim 5.0$), as well as three popular mouse pad materials, i.e., rubber ($\varepsilon = 2.3 \sim 3.0$), fabric ($\varepsilon = 1.5 \sim 3.0$) and leather ($\varepsilon = 1.5 \sim 2.5$).



(a) The left click circuit.

(b) The right click circuit.

Fig. 16. Impact of materials on IEMI-induced two-side clicks.



(a) Up.

(b) Down.

(c) Left.

(d) Right.

Fig. 17. Impact of materials on IEMI-induced movements in four target directions.

The thickness of each material is 3 mm. To simulate a real attack scenario, the mouse and the attack device are separated by "a table" or "a mouse pad and a table" with different material combinations (a total of 8 material combinations). The injection power is set to 4 W.

■ **Click.** For each material combination, we conduct 1600 IEMI attacks on the left- and right-click circuits of the mice and measure the corresponding click success rates and interference rates. As shown in Fig. 16, the click success rates (CSR) for each combination do not deteriorate noticeably (within 6.5%), compared with that without material obstruction. These results demonstrate that PuppetMouse can effectively pass through various materials and trigger IEMI-induced two-side clicks. Besides, the interference rates remain less than 3% under different materials, which proves the robustness of our click-movement interference elimination method against material changes.

■ **Movement.** For each material combination, We manipulate the mice to move in four target directions by IEMI injection. Particularly, we adjust the modulation frequency of the IEMI signals to make the ghost pattern drift in the target direction. For each target direction, we conduct 120 attacks and each attack lasts 5 seconds. The initial position of the mouse is at the center of the screen. Figure. 17 illustrates that the variation in the angular

deviations (ADs) for each target direction remains within 3.5 degrees across different materials, compared to those without material occlusion. This demonstrates the robustness of IEMI-induced movement against material variations. Furthermore, the interference rates under different materials remain lower than 2%, and the response rates are higher than 95.5%. This also validates the robustness of our click-movement interference elimination method when material changes. Hence, despite which of the above materials are used, PuppetMouse can still launch effective direction- and speed-adjustable movement attacks.

6.6 Case Study

We envision that there could be two types of potential real-world attack scenarios: (1) Screen-visible attack: the attacker can gain access to the victim's screen, which allows them to view its contents in real time. This type of attack can be achieved through various means, e.g., shoulder surfing, remote surveillance using compromised webcams, screen-sharing applications, or even utilizing telescopic lenses from a distance. In this case, attackers can monitor the behaviour of the cursor on the screen timely. (2) Screen-invisible attack: In this attack, the attacker is unable to directly view the victim's screen, but s/he can launch PuppetMouse attacks remotely. In the following, we will provide an example of each attack scenario and evaluate their effectiveness. Particularly, the target mouse and connected computer are Mouse Aigo Q21 and the Lenovo Xiaoxin Pro14 laptop. In each attack attempt, the initial position of the mouse is random.

6.6.1 Screen-visible Attack: Firewall Disabling. The attacker can stealthily monitor the laptop's screen and inject IEMI signals when the victim is not using the mouse. To disable the firewall on the laptop, the attacker needs to perform five mouse movements and six clicks, as detailed in Appendix. A.1. We invite three volunteers to launch this attack 30 times. As a result, the firewall of the laptop is disabled successfully in all 30 tests, and it takes about 147 seconds on average, with a maximum of 182 seconds and a minimum of 129 seconds. Thus, PuppetMouse can help an attacker disable the firewall easily.

6.6.2 Screen-invisible Attack: Malicious WiFi Connection. The major challenge of a screen-invisible attack is that the attacker cannot obtain the precise position of the cursor on the screen. To overcome this, the attacker can employ the following strategy. Before launching each attack, the attacker first moves the cursor to the bottom right corner of the screen for initialization. Then, the attacker uses another set of a mouse and a laptop that is the same as the victim mouse and laptop models to simulate the attack, design the IEMI signals, and estimate the duration of each attack. Screen-invisible attacks can be achieved by applying these estimated signal parameters to the laptop of the real victim.

Taking a malicious WiFi connection as an example, before IEMI injection, the attacker first sets up a malicious WiFi router with strong transmitting power around the victim's laptop to ensure that the malicious WiFi ranked high on the WiFi list [31]. To make the victim laptop connect to the malicious WiFi, the attacker needs to perform four mouse movements and three clicks, as detailed in Appendix. A.2. We invite three volunteers to perform 30 attack attempts. The results show that in 11 of 30 attempts, the laptop is connected to the malicious WiFi successfully. On average, each attempt takes 73 seconds. Hence, PuppetMouse still poses real threats to existing computer systems even if their screen is invisible.

7 COUNTERMEASURES

7.1 IEMI Shielding

A straightforward approach to mitigating IEMI injection is the use of a Faraday cage [11] to cover the mouse. For instance, manufacturers could consider replacing the material of the mouse shell with metal. Although the Faraday cage can effectively reduce the impact of PuppetMouse, its widespread practical application remains challenging due to two reasons. (1) The Faraday cage might hinder the convenience of mouse operation. Because metals,

commonly used for Faraday cages [11], often have a higher density than traditional mouse shell materials like plastic [3]. Moreover, the effectiveness of the Faraday cage is proportional to its thickness. Although increasing the thickness enhances the effectiveness of IEMI shielding, it also makes the mouse much heavier, thereby compromising the user experience. (2) More importantly, the Faraday cage could disrupt communication between mice (particularly wireless ones) and the connected computer [29]. This disruption significantly affects the normal operation of the mouse. An advanced strategy involves modifying the internal circuitry within the mouse to enhance resilience against IEMI interference. Key practices include using shielding and isolation layers to protect critical circuitry [54], designing the PCB layout to minimize coupling between IEMI signals and circuits [37], selecting components with robust electromagnetic compatibility (EMC) characteristics, and performing more rigorous EMC testing and validation during the design phase [52]. These proactive measures are essential for maintaining consistent and dependable performance of the mouse circuitry in the presence of electromagnetic disturbances.

7.2 External Force Detection

The primary distinction between human-manipulated clicks and IEMI-induced clicks is whether an external force is applied to the microswitch. Thus, another method to protect the mouse from IEMI-induced click is to detect external forces. Particularly, the manufacturers could integrate a pressure sensor (e.g., force sensing resistors [60]) into the microswitch circuit. When a user presses the microswitch, the resulting external force deforms the pressure sensor, changing its resistance or capacitance [49]. This deformation generates specific voltage signals in the microswitch circuit (called pressure signals). In this case, the MCU of the mouse can simultaneously detect the desired switch signal and pressure signal. In contrast, the IEMI-induced clicks only trigger the switch signals, but not the pressure signal. This distinction allows the mouse's MCU to distinguish between human-manipulated clicks and IEMI-induced clicks. Although this countermeasure can reduce the impact of IEMI on the microswitch circuit, it requires to modify the mouse hardware. Therefore, this countermeasure is not applicable to existing mice.

7.3 Motion Vector Analysis

Another method to defend against IEMI-induced movement is to calculate the average pixel movement of the image (i.e., the average motion vector). This is because the key distinction between human-manipulated movements and IEMI-induced movements is whether the differences they produce in successively captured images are global or local. In particular, human-manipulated movements cause global image differences since the entire image moves towards the target direction, whereas IEMI-induced movements result in local image differences (i.e., drifts of the ghost pattern). At the same movement speed, the average pixel movement generated by IEMI-induced movement is much smaller than that of human-manipulated movement. Here, we use the average motion vector to quantify the average pixel motion vector of the image [46]. Based on this phenomenon, we propose a defense mechanism: distinguishing between human-manipulated and IEMI-induced movements by calculating the average motion vectors of the images.

To validate our method, we invite three volunteers to move a Logitech G402 mouse to the right at a typical speed of 11 cm/s [50] for 5 seconds. During these experiments, we capture the images based on reverse engineering and further calculate the image motion vector for each experiment with the optical flow algorithm [46]. Each volunteer repeats the experiment 100 times. Meanwhile, we launch 10 IEMI attacks by injecting ghost patterns. To maintain consistent image movement speed, the ghost pattern drifts to the right at a speed of 0.5 pixels/frame. Each experiment also lasts for 5 seconds, and we calculate the corresponding image motion vector. The results show that when the mouse is manipulated by these three volunteers, the average image motion vectors are 0.37 pixels/frame, 0.32 pixels/frame, and 0.35 pixels/frame, respectively. During IEMI attacks, the average image motion

vector is approximately 0.004 pixels/frame. This significant difference indicates that the image motion vector during IEMI injection is indeed much smaller than that during manual mouse manipulation. Therefore, we can establish an empirical threshold (i.e., 0.2) for image motion vectors to distinguish between manual manipulation and IEMI manipulation. We find that the defense success rate can achieve 100%. It seems that this defense can completely prevent mice from IEMI intrusions. However, attackers could bypass it by enlarging the ghost pattern, for instance, making it cover 60% of the entire image. The trade-off of doing so is that the fake movement induced by IEMI can only be random. So, PuppetMouse still poses real threats to existing mouse-control systems.

7.4 Abnormal Signal Detection

When IEMI signals couple into a mouse's circuitry, voltage signals of the circuitry can exhibit abnormal changes (e.g., significant increases in amplitude [17] and frequency [54]). Therefore, real-time monitoring of stable circuit signals [17] under normal operating conditions (e.g., switch signals) can detect the presence of IEMI attacks effectively. Furthermore, the circuits preceding the microswitch and the image sensor can introduce filters [54] to further eliminate the voltage anomalies caused by IEMI signals, ensuring the normal operation of the mouse.

8 RELATED WORK

8.1 IEMI Attacks

With the development of wireless technology [20, 27], the risk of IEMI attacks is also increasing. These attacks disrupt or compromise the normal operation of devices by introducing malicious IEMI signals, such as injecting false commands [26, 39, 51, 53, 59], disrupting communication [33, 57, 62], and paralyzing the devices [22]. Generally, these attack methods intrude on either digital circuits or analog circuits.

■ **IEMI attacks on digital circuits.** Digital circuits are crucial in communications, enabling the processing and transmission of digital signals including encoding, modulation, demodulation, and decoding. Some existing works inject IEMI signals into digital circuits to block transmission signals [22], flip bits in signals [57, 62]and falsify frames [33], leading to the transmission of incorrect information during communication. Jang et al. [22] disrupt the communication channel between the IMU and the control unit, thereby paralyzing drones. Zhang et al. [62] and Xie et al. [57] inject IEMI interference to flip the bits of serial communication and inject malicious information into the serial communication system. Work in [33] falsifies CAN frames by tampering with sample points in the frame.

■**IEMI attacks on analog circuits.** Attackers can inject IEMI signals into analog circuits to disturb the output of sensors or manipulate the behavior of actuators. The works in [39, 53] employ IEMI to inject fake touch points into the touchscreen to achieve various basic touch events (e.g., taps and swipes). The attacks in [26, 59] exploit the inherent nonlinearity of the microphone to convert the injected IEMI signal into a baseband signal, allowing the microphone to receive inaudible commands. Tu et al. [51] deceive the temperature sensor by injecting IEMI signals to manipulate the control systems or circumvent temperature alarms. There are also some works targeting light sensors [38], magnetic speed sensors [45], smartphones [23], and CCD image sensors [25]. Furthermore, in works [13, 38, 61], attackers inject modulated IEMI signals into the system to disrupt actuators (e.g., power switches and servo). Zhang et al. [61] can bypass the controller by IEMI injection to directly control the actuator. Selvaraj et al. [38] use injected IEMI signals to manipulate the rotation angle of the servo. Dayanikli et al. [13] attack the power converter of the electric car, causing damage to the battery management system.

In this study, we introduce a new attack method that manipulates mouse clicks and movements by injecting IEMI signals into the analog circuits of image sensors and microswitches.

8.2 Mouse Attacks

As a vital input tool, the mouse plays an indispensable role in computer systems. However, it is also susceptible to potential security threats. Existing works targeting the mouse can be broadly categorized into three types according to attack vectors: computer-side attacks, computer-mouse communication attacks, and mouse-side attacks.

■ **Computer-side attacks.** These attacks typically inject malicious activities by malware on the system level against the mouse, including mouse hijacking [15], mouse phishing [43], and mouse monitoring [9]. Mouse Clickjacking [15] tricks users into clicking on a button or link on a different page than intended by using numerous transparent or opaque layers. The attack in [43] initiates malware when a user hovers the mouse over hyperlinked text and images that embed malicious links. Besides, the malware on the computer can also infer the privacy of the user by monitoring the behavior of the mouse [9]. However, these malware-based attacks exhibit some limitations. On one hand, they require significant overhead as attackers must successfully infiltrate the computer's system. On the other hand, the abnormal behavior induced by the malware may be detected by users or security software, frustrating the attack.

■ **Computer-mouse communication attacks.** These attacks usually engage in malicious activities against the mouse at the protocol level, encompassing both active tampering with the mouse's behavior [1, 42, 58] and passive monitoring [34, 35]. The works in [1, 42, 58] exploit communication protocol vulnerabilities by sending manipulated wireless signals to the target computer, impersonating legitimate mice for potential malicious activities. The works in [34, 35] sniff Bluetooth packets to reconstruct the trajectory of the mouse, compromising user privacy (e.g., passwords). However, these protocol-based attacks lack universality as they are only applicable to mice with specific protocols.

■ **Mouse-side attacks.** These attacks [8, 19, 28, 48] generally modify the hardware [28] or firmware [8, 19, 48] of the mouse and then load malicious code onto it, enabling the invasion of computers (e.g., hosts) connected to the mouse. However, the overhead of such attacks is significantly high, as the attackers need to gain full control over it for intrusive modifications. Simultaneously, this type of attack is prone to be detected by users.

Compared to the previous works, our attack demonstrates prevalence and non-invasiveness. Firstly, we target common components (i.e., the image sensor and microswitch) in all mice. This renders our attack protocol-independent, making it effective for both wired and wireless mice. Moreover, our method manipulates the mouse through IEMI signal injection without requiring any physical modifications to the mouse or infiltration into the computer's system. Consequently, our approach is non-intrusive and less prone to trigger user awareness.

9 DISCUSSION AND FUTURE WORK

■ **Attack Distance.** Currently, PuppetMouse can achieve good performance within an attack range of 5 cm. This is acceptable in real-world attack scenarios. In future work, there are several measures that can be adopted to enhance the attack distance and threat posed by PuppetMouse, which are also effective for other IEMI attacks. As analyzed in Section 6.4, sophisticated attackers may use more powerful injection equipment, such as high-gain directional antennas [21] and high-power RF amplifiers [47], to further improve the click success rate and extend the attack distance. To verify that signal power can enhance attack distance, we conduct an experiment. Specifically, we inject IEMI signals of different power levels (1 W, 2 W, 3 W, 4 W) into an HP M10 mouse and record the corresponding maximum attack distances. We observe that the maximum attack distances for different power levels are 0 cm, 1.5 cm, 4 cm, and 7 cm, respectively. The experimental results demonstrate a positive correlation between attack distance and signal power. In this case, if the attack device has sufficient power (e.g., 100 W), the attack range can reach up to 8 meters [22] Moreover, IEMI attack devices can be concealed within wireless chargers, which is a growing trend in consumer electronics. This camouflage enhances the covert nature of the attack.

■ **Improved Defensive Mechanism.** In Section 7, we explore several potential defense mechanisms against IEMI attacks. In our future work, we will focus on skillfully integrating the various defense methods proposed in Section 7 to create a more robust and comprehensive defense system. Additionally, we will continue to investigate the characteristic differences between injected IEMI signals and normal voltage signals and then develop innovative defense strategies. By exploring these defense methods, we aim to provide new insights into defending against similar IEMI attacks and enhance the ability of devices to withstand such threats.
■ **Impact of Human.** In our attack scenario, the attacker targets the mouse when it is not being manipulated. During this process, the IEMI signals affecting the mouse are directed straight towards it, without obstruction from the human body. Furthermore, compared to IEMI signals directly sent to the mouse, those reflected from the human have minimal impact on PuppetMouse attack. Therefore, surrounding humans have almost no influence on PuppetMouse attack.

If the attacker launches PuppetMouse attacks while a user is using the mouse, s/he can still achieve controllable two-side clicks. However, the IEMI signals cannot precisely control the direction and speed of mouse movement. This is because when the user moves the mouse, changes in the images captured by its image sensor are affected not only by ghost patterns but also by the surface beneath the mouse. Consequently, the mouse cannot rely solely on the drift of ghost patterns to determine its movement. In our future work, we plan to design a more sophisticated and adaptive ghost pattern by considering real-time analysis and feedback mechanisms for mouse movements. This advanced approach will enable us to achieve direction- and speed-adjustable movements even when the mouse is being manipulated by the user.
■ **Attack Improvement.** According to the experimental results in Section 6, we find that there is room for improvement in the performance of some mice regarding IEMI-induced clicks and movements. Currently, the performance of these mice may be limited by the power of the existing attack equipment or interference from other components (such as switch circuits). In our future work, we will further enhance the performance of PuppetMouse in two ways: first, we will use more powerful injection devices, such as high-gain directional antennas [21] and high-power RF amplifiers [47], to more effectively inject IEMI signals into the mice; secondly, we will design more refined IEMI signals to ensure that they only interfere with target components (i.e., microswitches and image sensors) without affecting non-target components (such as switch circuits).

10 CONCLUSION

In this paper, we introduce PuppetMouse, the first IEMI-based attack targeting mice to manipulate clicks and movements without physical contact. PuppetMouse crafts IEMI signals and injects them into the microswitches and image sensor of the mouse to achieve two-side clicks and direction- and speed-adjustable movements. By controlling the target mouse, PuppetMouse can execute a range of malicious activities on the connected computer to compromise personal privacy and property. Through extensive experiments and evaluation, we demonstrate that PuppetMouse is effective for most widely-used wired and wireless mice and resilient to attack distances and material obstructions. Furthermore, the case study shows that PuppetMouse is practical in real attack scenarios. To mitigate PuppetMouse's threats, we also discuss potential hardware- and software-based countermeasures. We hope that the PuppetMouse attack can raise awareness of the dangers posed by such IEMI attacks and encourage manufacturers and users to implement stronger protective measures.

ACKNOWLEDGMENTS

This paper is supported by the National Natural Science Foundation of China under grant U21A20462 and 62372400, "Pioneer" and "Leading Goose" R&D Program of Zhejiang under grant No. 2023C01033, and the Postdoctoral Fellowship Program of CPSF under Grant Number GZC20241488.

REFERENCES

[1] Mingrui Ai, Kaiping Xue, Bo Luo, Lutong Chen, Nenghai Yu, Qibin Sun, and Feng Wu. 2022. Blacktooth: Breaking through the Defense of Bluetooth in Silence. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security, CCS 2022, Los Angeles, CA, USA, November 7-11, 2022*, Heng Yin, Angelos Stavrou, Cas Cremers, and Elaine Shi (Eds.). ACM, 55–68. https://doi.org/10.1145/3548606.3560668

[2] GLOBAL CYBERSECURITY ALLIANCE. 2024. Industrial Automation and Control System Taxonomy. https://shorturl.at/dgtKM.

[3] AmesWeb. 2024. DENSITY OF MATERIALS. https://amesweb.info/Materials/Density-Materials.aspx.

[4] Amplifier. 2024. 4W 10-1000MHz RF power amplifier broadband RF power amplifier. https://www.ebay.com.sg/itm/124945225075.

[5] Antenna. 2024. 433MHZ 35dBi SMA High Gain Antenna Sucker Head Copper Signal Wireless Data. https://shorturl.at/twP29.

[6] Mohit Arora. 2011. *The art of hardware architecture: Design methods and techniques for digital circuits*. Springer Science & Business Media.

[7] USRP B210. 2024. USRP B210 SDR Kit - Dual Channel Transceiver (70 MHz - 6GHz). https://www.ettus.com/all-products/ub210-kit/.

[8] Brian Benchoff. 2014. A REAL MALWARE IN A MOUSE. https://hackaday.com/2014/03/31/a-real-malware-in-a-mouse/.

[9] Becky Bracken. 2023. Malware Uses Trigonometry to Track Mouse Strokes. https://www.darkreading.com/application-security/malware-uses-trigonometry-to-track-mouse-strokes.

[10] Roberto Brunelli. 2009. *Template matching techniques in computer vision: theory and practice*. John Wiley & Sons.

[11] S Jonathan Chapman, David P Hewett, and Lloyd N Trefethen. 2015. Mathematics of the Faraday cage. *Siam Review* 57, 3 (2015), 398–417.

[12] EUROPEAN COMMISSION. 2016. MEDICAL DEVICES: Guidance document. https://shorturl.at/abyBR.

[13] Gökçen Yilmaz Dayanikli, Rees R. Hatch, Ryan M. Gerdes, Hongjie Wang, and Regan Zane. 2020. Electromagnetic Sensor and Actuator Attacks on Power Converters for Electric Vehicles. In *2020 IEEE Security and Privacy Workshops, SP Workshops, San Francisco, CA, USA, May 21, 2020*. IEEE, 98–103. https://doi.org/10.1109/SPW50608.2020.00032

[14] IBE Electronics. 2022. PCB Shielding-type, material and purpose. https://www.pcbaaa.com/pcb-shielding-type-material-and-purpose/.

[15] Joe Elenjickal. 2022. What Is Clickjacking? How Can A Hacker Steal Your Mouse Clicks? https://prophaze.com/web-application-firewall/what-is-clickjacking/.

[16] Jack G Ganssle. 2004. A guide to debouncing. *Guide to Debouncing, Ganssle Group, Baltimore, MD, US* (2004), 1–22.

[17] Ming Gao, Fu Xiao, Weiran Liu, Wentao Guo, Yangtao Huang, Yajie Liu, and Jinsong Han. 2023. Expelliarmus: Command cancellation attacks on smartphones using electromagnetic interference. In *IEEE INFOCOM 2023-IEEE Conference on Computer Communications*. IEEE, 1–10.

[18] Andy Greenberg. 2016. A Hacker Tried to Poison a Florida City's Water Supply, Officials Say. https://www.wired.com/story/oldsmar-florida-water-utility-hack/.

[19] Roger Grimes. 2011. Yes, even a mouse can infect your network. https://www.csoonline.com/article/548716/insider-threats-yes-even-a-mouse-can-infect-your-network.html.

[20] Linqing Gui, Wenyang Yuan, and Fu Xiao. 2023. CSI-based passive intrusion detection bound estimation in indoor NLoS scenario. *Fundamental Research* 3, 6 (2023), 988–996.

[21] National Instruments. 2022. LP0410 Log Periodic PCB Antenna. https://www.ettus.com/all-products/lp0410/.

[22] Joon-Ha Jang, ManGi Cho, Jaehoon Kim, Dongkwan Kim, and Yongdae Kim. 2023. Paralyzing Drones via EMI Signal Injection on Sensory Communication Channels. In *30th Annual Network and Distributed System Security Symposium, NDSS 2023, San Diego, California, USA, February 27 - March 3, 2023*. The Internet Society. https://www.ndss-symposium.org/ndss-paper/paralyzing-drones-via-emi-signal-injection-on-sensory-communication-channels/

[23] Chaouki Kasmi and Jose Lopes Esteves. 2015. IEMI Threats for Information Security: Remote Command Injection on Modern Smartphones. *IEEE Transactions on Electromagnetic Compatibility* 57, 6 (2015), 1752–1755. https://doi.org/10.1109/TEMC.2015.2463089

[24] Paul Kinsler. 2020. Faraday's law and magnetic induction: cause and effect, experiment and theory. *Physics* 2, 2 (2020), 150–163.

[25] Sebastian Köhler, Richard Baker, and Ivan Martinovic. 2022. Signal Injection Attacks against CCD Image Sensors. In *ASIA CCS '22: ACM Asia Conference on Computer and Communications Security, Nagasaki, Japan, 30 May 2022 - 3 June 2022*, Yuji Suga, Kouichi Sakurai, Xuhua Ding, and Kazue Sako (Eds.). ACM, 294–308. https://doi.org/10.1145/3488932.3497771

[26] Denis Foo Kune, John D. Backes, Shane S. Clark, Daniel B. Kramer, Matthew R. Reynolds, Kevin Fu, Yongdae Kim, and Wenyuan Xu. 2013. Ghost Talk: Mitigating EMI Signal Injection Attacks against Analog Sensors. In *2013 IEEE Symposium on Security and Privacy, SP 2013, Berkeley, CA, USA, May 19-22, 2013*. IEEE Computer Society, 145–159. https://doi.org/10.1109/SP.2013.20

[27] Wengang Li, Yaqin Xu, Chenmeng Zhang, Yiheng Tian, Mohan Liu, and Jun Huang. 2023. Multi-Frequency-Ranging Positioning Algorithm for 5G OFDM Communication Systems. *Chinese Journal of Electronics* 32, 4 (2023), 773–784. https://doi.org/10.23919/cje.2021.00.124

[28] Matt Liebowitz. 2011. Hackers Use Rogue Mouse to Crack Firewall. https://www.nbcnews.com/id/wbna43568622.

[29] Tiantian Liu, Feng Lin, Zhangsen Wang, Chao Wang, Zhongjie Ba, Li Lu, Wenyao Xu, and Kui Ren. 2023. MagBackdoor: Beware of Your Loudspeaker as a Backdoor for Magnetic Injection Attacks. In *2023 IEEE Symposium on Security and Privacy (SP)*. IEEE, 3416–3431.

[30] magalsh64. 2023. Hack to turn a Logitech G402 Mouse into a Camera! https://github.com/maglash64/g402_camera.
[31] Microsoft. 2023. Understanding and configuring Windows Connection Manager. https://learn.microsoft.com/en-us/windows-hardware/drivers/mobilebroadband/understanding-and-configuring-windows-connection-manager.
[32] Newsroom. 2023. New Financial Malware 'JanelaRAT' Targets Latin American Users. https://thehackernews.com/2023/08/new-financial-malware-janelarat-targets.html.
[33] Hiroto Ogura, Ryunosuke Isshiki, Kengo Iokibe, Yuta Kodera, Takuya Kusaka, and Yasuyuki Nogami. 2020. Electrical Falsification of CAN Data by Magnetic Coupling. In *2020 35th International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC)*. 348–353.
[34] Xian Pan, Zhen Ling, Aniket Pingley, Wei Yu, Kui Ren, Nan Zhang, and Xinwen Fu. 2013. How Privacy Leaks From Bluetooth Mouse?. In *20th Annual Network and Distributed System Security Symposium, NDSS 2013, San Diego, California, USA, February 24-27, 2013*. The Internet Society. https://www.ndss-symposium.org/ndss2013/how-privacy-leaks-bluetooth-mouse
[35] Xian Pan, Zhen Ling, Aniket Pingley, Wei Yu, Nan Zhang, Kui Ren, and Xinwen Fu. 2016. Password Extraction via Reconstructed Wireless Mouse Trajectory. *IEEE Trans. Dependable Secur. Comput.* 13, 4 (2016), 461–473. https://doi.org/10.1109/TDSC.2015.2413410
[36] Electronic Products. 2011. Optical mouse technology: Here to stay, still evolving. https://www.electronicproducts.com/optical-mouse-technology-here-to-stay-still-evolving/.
[37] Riverdi. 2020. Mitigating Electromagnetic Interference in Display Technologies Across Sectors: A Comprehensive Guide. https://riverdi.com/blog/mitigating-electromagnetic-interference-in-display-technologies-across-sectors.
[38] Jayaprakash Selvaraj, Gökçen Yilmaz Dayanikli, Neelam Prabhu Gaunkar, David Ware, Ryan M. Gerdes, and Mani Mina. 2018. Electromagnetic Induction Attacks Against Embedded Systems. In *Proceedings of the 2018 on Asia Conference on Computer and Communications Security, AsiaCCS 2018, Incheon, Republic of Korea, June 04-08, 2018*, Jong Kim, Gail-Joon Ahn, Seungjoo Kim, Yongdae Kim, Javier López, and Taesoo Kim (Eds.). ACM, 499–510. https://doi.org/10.1145/3196494.3196556
[39] Haoqi Shan, Boyi Zhang, Zihao Zhan, Dean Sullivan, Shuo Wang, and Yier Jin. 2022. Invisible Finger: Practical Electromagnetic Interference Attack on Touchscreen-based Electronic Devices. In *43rd IEEE Symposium on Security and Privacy, SP 2022, San Francisco, CA, USA, May 22-26, 2022*. IEEE, 1246–1262. https://doi.org/10.1109/SP46214.2022.9833718
[40] Shreyas Sharma. 2024. Understanding PCB Thickness: A Comprehensive Guide. https://www.wevolver.com/article/pcbs-thickness-understanding-thickness-variations.
[41] Joseph A Shaw. 2013. Radiometry and the Friis transmission equation. *American journal of physics* 81, 1 (2013), 33–37.
[42] Matt Sheimo. 2021. AHHH! MY MOUSE AND KEYBOARD WERE HACKED! https://www.sikich.com/insight/ahhh-my-mouse-and-keyboard-were-hacked/.
[43] Kelly Sheridan. 2017. New Attack Method Delivers Malware Via Mouse Hover. https://www.darkreading.com/endpoint-security/new-attack-method-delivers-malware-via-mouse-hover.
[44] Sol Sherr. 2012. *Input devices*. Vol. 1. Elsevier.
[45] Yasser Shoukry, Paul D. Martin, Paulo Tabuada, and Mani B. Srivastava. 2013. Non-invasive Spoofing Attacks for Anti-lock Braking Systems. In *Cryptographic Hardware and Embedded Systems - CHES 2013 - 15th International Workshop, Santa Barbara, CA, USA, August 20-23, 2013. Proceedings (Lecture Notes in Computer Science, Vol. 8086)*, Guido Bertoni and Jean-Sébastien Coron (Eds.). Springer, 55–72. https://doi.org/10.1007/978-3-642-40349-1_4
[46] Wesley E Snyder and Hairong Qi. 2017. *Fundamentals of computer vision*. Cambridge University Press.
[47] Empower RF System. 2022. High Power RF Amplifiers Systems and Modules. https://www.empowerrf.com/products/rf-power-amplifier.php.
[48] Mike Szczys. 2014. BADUSB MEANS WE'RE ALL SCREWED. https://hackaday.com/2014/10/05/badusb-means-were-all-screwed/.
[49] Duane Tandeske. 1990. *Pressure sensors: selection and application*. CRC Press.
[50] Ulrich Tränkle and Detlef Deutschmann. 1991. Factors influencing speed and precision of cursor positioning using a mouse. *Ergonomics* 34, 2 (1991), 161–174.
[51] Yazhou Tu, Sara Rampazzi, Bin Hao, Angel Rodriguez, Kevin Fu, and Xiali Hei. 2019. Trick or Heat?: Manipulating Critical Temperature-Based Control Systems Using Rectification Attacks. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security, CCS 2019, London, UK, November 11-15, 2019*, Lorenzo Cavallaro, Johannes Kinder, XiaoFeng Wang, and Jonathan Katz (Eds.). ACM, 2301–2315. https://doi.org/10.1145/3319535.3354195
[52] Vicon. 2024. WHAT IS ELECTROMAGNETIC COMPATIBILITY TESTING ? https://www.vicom.com.au/page/217/emc-emi-testing-what-is-electromagnetic-compatibility-testing-.
[53] Kai Wang, Richard Mitev, Chen Yan, Xiaoyu Ji, Ahmad-Reza Sadeghi, and Wenyuan Xu. 2022. GhostTouch: Targeted Attacks on Touchscreens without Physical Touch. In *31st USENIX Security Symposium, USENIX Security 2022, Boston, MA, USA, August 10-12, 2022*, Kevin R. B. Butler and Kurt Thomas (Eds.). USENIX Association, 1543–1559. https://www.usenix.org/conference/usenixsecurity22/presentation/wang-kai
[54] Kai Wang, Richard Mitev, Chen Yan, Xiaoyu Ji, Ahmad-Reza Sadeghi, and Wenyuan Xu. 2024. Analyzing and Defending GhostTouch Attack against Capacitive Touchscreens. *IEEE Transactions on Dependable and Secure Computing* (2024).

[55] wiki. 2024. Computer mouse. https://en.wikipedia.org/wiki/Computer_mouse.
[56] WiKi. 2024. Optical mouse. https://en.wikipedia.org/wiki/Optical_mouse.
[57] Zhixin Xie, Chen Yan, Xiaoyu Ji, and Wenyuan Xu. 2023. BitDance: Manipulating UART Serial Communication with IEMI. In *Proceedings of the 26th International Symposium on Research in Attacks, Intrusions and Defenses, RAID 2023, Hong Kong, China, October 16-18, 2023*. ACM, 63–76. https://doi.org/10.1145/3607199.3607249
[58] Fenghao Xu, Wenrui Diao, Zhou Li, Jiongyi Chen, and Kehuan Zhang. 2019. BadBluetooth: Breaking Android Security Mechanisms via Malicious Bluetooth Peripherals. In *26th Annual Network and Distributed System Security Symposium, NDSS 2019, San Diego, California, USA, February 24-27, 2019*. The Internet Society. https://www.ndss-symposium.org/ndss-paper/badbluetooth-breaking-android-security-mechanisms-via-malicious-bluetooth-peripherals/
[59] Zhifei Xu, Runbing Hua, Jack Juang, Shengxuan Xia, Jun Fan, and Chulsoon Hwang. 2021. Inaudible Attack on Smart Speakers With Intentional Electromagnetic Interference. *IEEE Transactions on Microwave Theory and Techniques* 69, 5 (2021), 2642–2650. https://doi.org/10.1109/TMTT.2021.3058585
[60] SI Yaniger. 1991. Force sensing resistors: A review of the technology. *Electro International, 1991* (1991), 666–668.
[61] Youqian Zhang and Kasper Rasmussen. 2022. Detection of Electromagnetic Signal Injection Attacks on Actuator Systems. In *25th International Symposium on Research in Attacks, Intrusions and Defenses, RAID 2022, Limassol, Cyprus, October 26-28, 2022*. ACM, 171–184. https://doi.org/10.1145/3545948.3545949
[62] Youqian Zhang and Kasper Rasmussen. 2023. Electromagnetic Signal Injection Attacks on Differential Signaling. In *Proceedings of the 2023 ACM Asia Conference on Computer and Communications Security, ASIA CCS 2023, Melbourne, VIC, Australia, July 10-14, 2023*, Joseph K. Liu, Yang Xiang, Surya Nepal, and Gene Tsudik (Eds.). ACM, 314–325. https://doi.org/10.1145/3579856.3590326

A DETAILS IN CASE STUDY

In the following, we provide detailed descriptions of steps that an attacker would take to conduct these two malicious activities in Sec.6.6, i.e., disabling the firewall and connecting to a malicious WiFi.

A.1 Firewall Disabling

To disable the firewall of the laptop, the attacker needs to perform a series of operations:

1) move the mouse to “Start menu” and click on it;
2) move the mouse to “Control Panel” and click on it;
3) move the mouse to “System and Security” and click on it;
4) click on “Windows Defender Firewall”;
5) move the mouse to “Turn Windows Defender Firewall on or off” and click on it;
6) move the mouse to and click on "Turn off".

A.2 Malicious WiFi Connection

To connect to the malicious WiFi, the attacker needs to perform a series of operations:

1) move the mouse to the bottom right corner of the screen for initialization;
2) move the mouse from the bottom right corner of the screen to the WiFi icon in the taskbar and click on it;
3) move and click on "Manage WiFi connections";
4) move and click on the malicious WiFi that ranks high (e.g., the first) in the WiFi list.