2024 IEEE Symposium on Security and Privacy (SP) | 979-8-3503-3130-1/24/$31.00 ©2024 IEEE | DOI: 10.1109/SP54263.2024.00231

# From Virtual Touch to Tesla Command: Unlocking Unauthenticated Control Chains From Smart Glasses for Vehicle Takeover

Xingli Zhang*[1] Yazhou Tu*[2] Yan Long[3] Liqun Shan[1,6] Mohamed A Elsaadani[1]
Kevin Fu[4] Zhiqiang Lin[5] Xiali Hei[1,6]
[1]University of Louisiana at Lafayette [2]Auburn University [3]University of Michigan
[4]Northeastern University [5]The Ohio State University [6]University of Pennsylvania

*Abstract*—**This paper studies vulnerabilities at the intersection of wearable devices and automated control systems. Particularly, we focus on exploiting smart glasses as an entry point and unveil the threats of taking over security-critical automated control chains without user verification or interaction. These vulnerabilities can be especially pertinent in scenarios where security mechanisms only depend on entry point security with minimal user verification (relying on complete trust over previous nodes in automated control chains). We have validated the effects of our attacks on real-world systems (e.g., Tesla vehicles) that are controlled by software and automation tools such as Apple Shortcuts or IFTTT. We show how our contactless, speaker-independent, and electromagnetic interference based attacks can control functionalities such as unlocking doors and initiating remote start of Tesla vehicles, even though the victim's phone is in a lock-screen status. Our findings not only demonstrate the potential for unauthorized control over automated, connected systems but also highlight the urgent need for more robust security measures in the integration of wearable technology with broader automation frameworks.**

## 1. Introduction

The pervasive integration of mobile and wearable technology, smart home devices, and connected vehicles is redefining our interaction with the physical world. Particularly, control chains over these real-world systems via automation tools such as Apple Shortcuts [1] and IFTTT [2] are rapidly emerging [3], [4], [5], [6]. This trend has been supported by service providers and manufacturers that are improving their products' functionalities and readily accessibility to end users, through developing their APIs and integrating with popular automation tools. A notable instance of this progression is Tesla's recent adoption of official support for Apple Shortcuts in August 2023 [7], [8], and their subsequent release of official APIs in October 2023 [9], [10], indicating a significant shift towards more interconnected and automated systems.

Instead of manually opening apps for individual subtasks, users can utilize automated control chains to execute tasks ranging from smart home device management to vehicle control with unprecedented convenience and efficiency. The activation of these chains is often initiated by user interactions to invoke functionalities across different modules and perform various tasks. In this paper, we investigate the vulnerabilities at the intersection of wearable devices and automated control systems. We dive into the potential existence of "weakest links" within these interconnected systems which, if exploited by adversaries, might enable taking over automated control chains without the necessity of verification, physical contact, or the use of sensitive user-specific information.

Smart glasses, a rapidly growing segment of the wearable technology market, are increasingly becoming integral to our daily lives due to their diverse features and functionalities. They are anticipated to see significant market growth [11], [12] and include more diverse technologies in the near future [13], [14], [15]. Once connected to a user's phone via Bluetooth, smart glasses can serve as a gateway for users to interact with automated systems.

Our research explores risks inherent in automated control chains (Fig. 1) and focuses on exploiting the wearable gateway provided by smart glasses to control security-critical systems. First, we reveal how adversaries can utilize smart glasses as an entry point through intentional electromagnetic interference (EMI) attacks to remotely trigger touch activation of voice assistant (VA) systems with low-cost attack devices. The activation process does not require physically touching the victim's device or relying on the owner's voice. Our attack approach targets devices that do not have always-on microphones, like smart glasses, that cannot therefore be directly triggered by acoustic signals.

We then extend our validation of the threats to a variety of control chains. Specifically, we investigate the attacks on chains officially supported by Apple Shortcuts and Tesla, and those involving third-party service providers like IFTTT and Tessie. These chains enable control over a broad spectrum of Tesla functionalities [6], [16]. The typical structure of these targeted control chains, as depicted in Fig. 1, includes wearable devices, smartphones, VA systems, automation tools, apps such as Tesla, and servers of service providers and manufacturers, ultimately controlling physical systems like vehicles. After contactlessly activating the victim's VA system, we inject phrases to invoke automation

. * Both authors contributed equally.

DOI 10.1109/SP54263.2024.00182

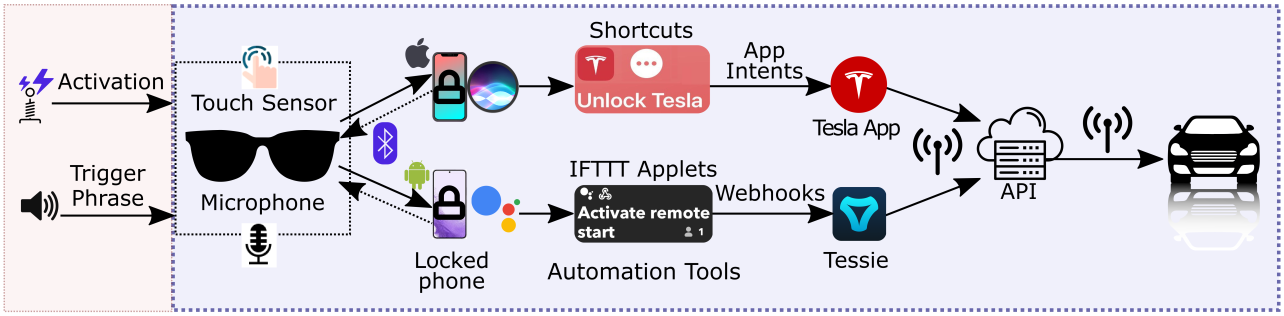


Figure 1: In automated control chains from wearable devices to vehicles, the subsequent components implicitly trust their previous nodes to execute tasks. Our study focuses on exploiting the wearable gateway (smart glasses) and exploring the effects of the attacks in control chains based on automation tools.

tools' actions using artificial intelligence (AI) speech signals. We demonstrate successful end-to-end attacks that require no verification or user interaction when the victim's phone remains in a lock-screen status. Our discovery highlights a critical oversight: once modules in automated control chains are installed and configured, they tend to trust previously established modules implicitly, even when these control chains are triggered by adversaries using physical signals independent of the victim user.

Prior efforts have focused on the security of voice assistants [17], [18], [19], [20] on devices such as smartphones and smart speakers that are equipped with always-on microphones, but not in the context of complex automated control chains like those we examine in Fig. 1. Our work studies the security implications of wearable devices within these chains, and demonstrates how contactless, EMI-based attacks can affect such chains to manipulate critical systems without requiring sensitive user-specific information such as their voice [21], [22] and fingerprint [23], [24].

Our research highlights that emerging automation tools (e.g., Shortcuts and IFTTT) based systems are built on a foundation of transitive trust. With the likelihood of more vehicles and smart home systems being operated through automation tools, human interaction with these systems becomes increasingly security-sensitive. The potential attack vector could extend beyond Tesla vehicles to other systems that offer API access and are integrated with automation tools [25]. Thus, there is a growing need for user-centered security mechanisms to improve trust in automated control chains while maintaining usability and accessibility.

We summarize the contributions of this paper as follows:

- We present the first study on the security of automated control chains (Fig. 1) under the effects of physical signal injection attacks. Our study reveals the vulnerabilities at the intersection of wearable devices and automated control systems, highlighting the threats of unauthorized control over security-critical functionalities of real-world automated systems.
- We propose an attack approach that exploits the wearable gateway (smart glasses) as the entry point to manipulate automated control chains with contactless, speaker-independent, and EMI-based attacks. We analyze the attacks with different platforms, apps, and voice assistants in the control chains.
- We explore the risks by validating the attacks on one type of the most safety-critical systems – Tesla vehicles. We show how the attacks can control functionalities of the automated, connected cars[1].

## 2. Background

### 2.1. Smart Glasses

Smart glasses are increasingly popular wearable devices that offer users non-intrusive audio and integration of diverse functionalities into daily activities [11], [13], [15], [26]. They can come with variations such as sunglasses, decorative glasses, and sport glasses [26], [27], [28]. They are designed to connect to smartphones or tablets via Bluetooth, and a typical structure of a popular model, the Razer Anzu, is illustrated in Fig. 2.

Smart glasses are typically equipped with speakers, microphones, and touch sensors. The speakers, often located on the frames, provide open-ear audio, making the devices favored options for individuals with hearing impairments. They also prevent the discomfort and potential infections associated with in-ear devices and help users remain alert to environmental sounds compared to earphones.

The microphones in smart glasses are often omnidirectional but are not always active, a design choice made for energy conservation. As a result, these microphones cannot be directly activated by acoustic signals or other carriers that induce signals in microphone circuits [17], [19], [29]. The microphones are used for voice interactions between the paired smartphone and the person who wears it.

Most smart glasses feature touch-responsive control on one or both temples. Such control enables users to change music tracks, play or pause media, manage calls, and activate the smartphone's voice assistant. Touch control has

[1] Demos of the proof-of-concept attacks are available at https://tinyurl.com/wmrr6u48

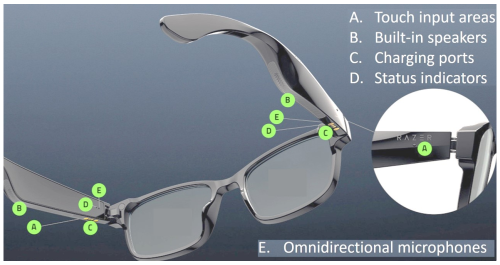
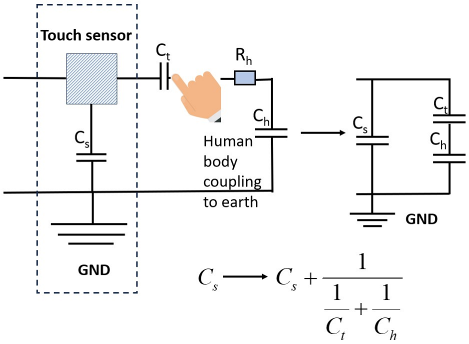


Figure 2: The typical frame structure of smart glasses [40].



$$C_s \longrightarrow C_s + \frac{1}{\frac{1}{C_t} + \frac{1}{C_h}}$$

Figure 3: A finger touch model of self capacitive touch sensor.

emerged as one of the most prevalent human-machine interaction interfaces in many applications [30], [31], [32], [33].

Compared to other touch-sensing technologies, capacitive touch sensing has become an industry standard among mobile device manufacturers [34]. A typical capacitive touch sensor is composed of an insulator, like plastic, coated with a conductive material, such as indium tin oxide (ITO) [35]. The conductive material, as an electrode, forms a capacitor with the earth ground or with the other electrode. They are referred to as self and mutual capacitance, respectively [36]. The touch functionality on smart devices often uses a self-capacitive touch sensor, as it is simpler to implement and provides accurate touch detection [37], [38]. An approaching finger introduces extra capacitance, and the controller detects changes in capacitance between the earth ground and the electrode to identify touch events. As shown in Fig. 3, in the equivalent circuit diagram (right), touch sensor capacitance ($C_t$) and human body capacitance ($C_h$), are parallel with the original capacitor ($C_s$). They increase the total capacitance measured by the sensor and further affect the digital signal. This change will be measured by the controller of the touch sensor. If the change reaches a particular threshold, a finger presence is flagged by the controller [39].

## 2.2. Automated Control Chains

Automation tools [41] like Shortcuts and IFTTT have revolutionized task management and execution with their automated "IF This, Then That" action chains. These tools simplify the connection process for repetitive tasks, streamlining workflow automation and enhancing productivity.

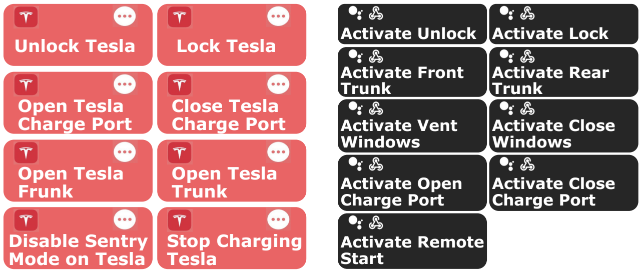


Figure 4: Security-sensitive Tesla Shortcuts (left) and IFTTT applets (right).

**Shortcuts.** Originally known as Workflow, Shortcuts is a visual scripting app developed for iOS, iPadOS, macOS, and watchOS. This native application empowers users to create macros or personalized shortcuts, automating specific tasks on their devices. These custom sequences, which can be shared online via iCloud, streamline workflow processes, especially for repetitive tasks. With the introduction of iOS 13.1, Shortcuts gained enhanced automation capabilities, offering system-suggested shortcuts based on user routines and app interactions [42]. These shortcuts, which can involve multiple steps from various apps, can be quickly initiated via Siri. An example is the "Open Tesla Trunk/Frunk" voice command, allowing operations without the need to search for keys or open the app manually. Fig. 4 shows security-critical commands for Tesla.

Users can activate Shortcuts either by tapping the widgets within the app, speaking to Siri (by voice command "Hey Siri, [name of shortcut]"), or by automating them based on set specific events like time or location changes. This feature is also available for manual installation on Android phones, with shortcuts being automatically generated for iOS users as they utilize certain apps (Fig. 4).

**IFTTT.** Similar to Shortcuts, IFTTT is an online digital automation platform that connects numerous IoT devices, applications, and websites working with each other through applets. In the automated control chains involving car APIs such as Tessie, IFTTT generates a Webhook request to Tessie and attaches the command and token information; Tessie will then communicate with the Tesla API to interact with the user's car. IFTTT empowers users to control their devices in a significantly more efficient way. IFTTT has more than 650 services available to users [43], [44]. Other automation tools, such as Microsoft Power Automate, can be combined to support more powerful interconnection and automation.

**APIs of Automobile.** Vehicle and smart home manufacturers are providing APIs that allow more computer programs to access their systems with authentication.

The first type of API is the official public API. Recently, Tesla has officially released its API documentation to third-party apps [10]. To use the APIs, users with Tesla accounts

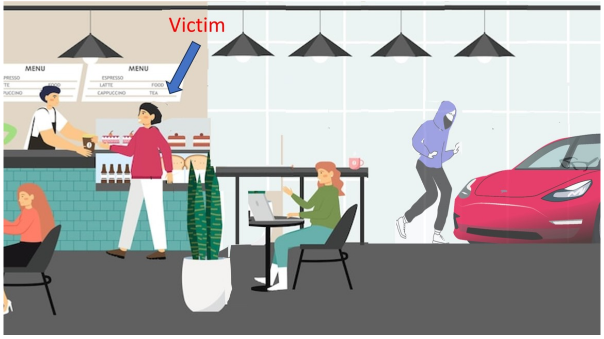


Figure 5: An example of the attack scenario. The victim temporarily parks his/her car outside a coffee shop and leaves their smart glasses inside the car. The adversary performs the attack in his/her absence.

first submit access requests. After being approved and obtaining a Client ID and Client Secret for their app, they can use these credentials to get a user Access Token with OAuth 2.0 authentication. Access Tokens are used to authenticate requests that provide private user account information or perform actions [9].

The second type is 3rd-party APIs. For example, Tessie is a popular paid 3rd-party software to interact with Tesla cars. It communicates with Tesla to log a vehicle's data automatically via Tesla's API and also sends control commands to Tesla [45]. The users are required to link their Tesla account with this app. After successful authentication, Tessie talks to Tesla to get data for the corresponding car. For further use of IFTTT, an access token is generated in Tessie and the IFTTT applet will execute a Webhook to access Tessie's APIs using the token [16].

Further, manufacturers' APIs can be reverse-engineered by analyzing the apps provided by the manufacturers. Many different vehicle systems (e.g., Drone Mobile, Audi, etc.) have their APIs. The open-source versions of their unofficial APIs can be controlled with programs by advanced users or service providers. Cars belong to one type of the most security-critical cyber-physical systems that are increasingly connected and automated. While connected cars are becoming an unstoppable trend [9], [10], [46], their security impacts still need thorough investigations [47].

## 3. Threat Model

The adversary aims to exploit a victim's smart glasses to manipulate the victim's vehicle, which is controlled remotely by the automated control chains.

**Adversary Capabilities.** We assume that the adversary cannot directly contact the victim's devices, including smart glasses and smartphones. The adversary also cannot tamper with the software of the victim's devices, such as running malicious codes to compromise the system or changing any original settings. The attack does not require the victim to unlock the smartphone or verify biometrics. We make no assumptions about acquiring the victim's voice or fingerprint.

Although the attackers have no direct access to the targeted smart glasses, they are aware of the characteristics of the glasses depending on the specific manufacturer and model, especially the activation methods. The maker and model can be easily observed, as most smart glass manufacturers print their brand name or logo on the frame. Therefore, attackers could obtain prior knowledge of the device by analyzing the same or similar commercial-off-the-shelf products.

**Attack Scenarios.** The adversary performs EMI-based activation by attacking smart glasses without touching them. Subsequently, the adversary can play trigger phrases with a speaker or use specialized devices [17], [18], [20], [48] to inject audio signals to the microphone of smart glasses without being noticed by the user. We assume that the victim temporarily takes off his/her smart glasses. For instance, a user may leave smart glasses nearby, such as a table or shelf, in a public area. Additionally, since the space inside a locked car is considered a private space, a user may leave his/her smart glasses inside the car after parking (as illustrated in Fig. 4). The user's mobile phone is still within the Bluetooth communication range to pair with the smart glasses [2], and this connection is kept on at least 6 to 10 seconds (Table 2) after the attack starts. During the attack, the victim does not have to be close to his/her Tesla vehicle because it can be remotely accessed by Tesla's Server once the automated control chains are executed.

The smart glasses will usually be left in the "on" mode if the driver does not intentionally long-press the sensing zone on the template or fold the template. For convenience, users usually will not turn it off, as sometimes they will not take a long leave, and the device is power-saving. After returning to the car, they do not need to turn on the glass again. We assume that the owner is not intensively focusing on the phone screen during the short period of the attack and could be walking, eating a meal, having conversations with friends, etc. This is a common scenario assumed by previous interactive attacks against smartphones [49], [50] since people are not always interacting with their smartphones. A concrete attack scenario is illustrated in Fig. 5.

**Device Characteristics.** Siri is the default VA system on iPhones, while Google Assistant is usually the one on Android phones. If the victim uses an iPhone, we assume that the iPhone keeps its default setting "Allow Siri When Locked". It is assumed that the victims do not delete the Shortcuts from their iPhones and they have interacted with Tesla via Shortcuts at least once. For Android devices, we assume that victims have set up automation tools. To do so, users have installed apps such as IFTTT, enabled Google Assistant when locked, and turned on personal results on headphones (this setting is used to control what kind of info Google Assistant can say to users in headphones [51]). The victims have configured IFTTT applets based on online instructions provided by Tessie.

[2] Our experiments observed that the range for typical smart glasses reached over 70 m.

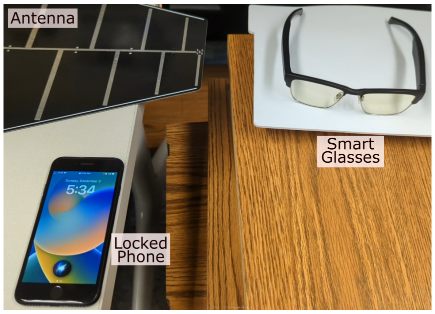


Figure 6: Experiment settings of smart glasses under EMI attacks with an antenna. We observe that EMI signals can contactlessly activate VA systems on phones paired with smart glasses.

## 4. Attack Methodology

In this section, we explore how to contactlessly activate the target phone's VA through smart glasses and invoke automated actions to control Tesla functionalities without having to unlock the phone.

### 4.1. Contactless Speaker-Agnostic VA Activation

Mainstream commercial smart glasses utilize capacitive touch sensors to detect user touch inputs. For example, Razer Anzu smart glasses measure the 2-second continuous press to activate the voice assistant system on the user's paired smartphone, while some other smart glasses detect quick double taps for the same action.

Adversaries cannot directly wake up smart glasses' VA system by affecting their microphones, since they are usually off unless they are turned on by a user's app for recording. Our work discovered the feasibility of using smart glasses as the entry point to manipulate automated control chains by targeting the touch sensor of smart glasses with EMI.

Researchers observed that strong electromagnetic signals can cause changes in electric charge in capacitive touchscreen controllers, and the changes will be detected as a false touch [49]. Shan et al. utilized electrode plates to generate electric fields, which affected the output voltage variation of the charge transfer (QT) sensor in a touchscreen controller and induced ghost touches [52]. These previous studies [49], [52], [53] demonstrated the feasibility of using near-field EMI to affect capacitive touch sensing and manipulated the graphical user interface events of smartphone screens. Additionally, researchers studied EMI attacks to alter the measurements of a single sensor circuit [29], [54], [55], [56]. In these scenarios, the EMI signals entered the analog circuit via backdoor coupling, and exploited non-linearity in sensor circuit components or microcontroller pins to induce specific changes in the detected voltage to manipulate sensor data [54], [57], [58], [59], [60]. Inspired by prior works, here we investigate whether EMI attacks can be applied to the touch sensors of smart glasses to activate the voice assistants.

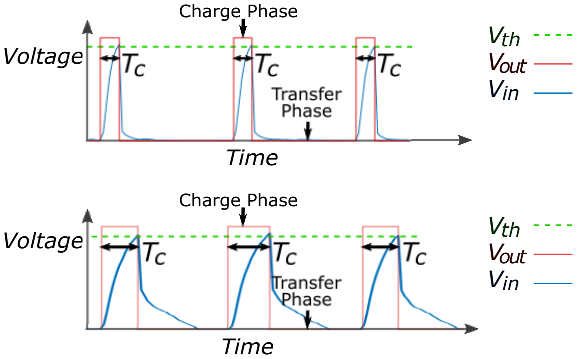


Figure 7: An illustration of capacitive touch sensing principle without (top) and with (bottom) touch. In the charge phase, $V_{out}$ becomes a high voltage. To estimate the capacitance, the circuit measures the charge time $T_C$ by comparing the input voltage $V_{in}$ to a threshold $V_{th}$. In the transfer phase, $V_{out}$ becomes a low voltage (e.g., zero) to discharge the circuit. Compared to when there is no touch, $T_C$ is usually longer when a human user touches the electrode.

**Observations.** To investigate the effects of EMI on smart glasses, we test Eyewear Plus smart glasses and scan EMI frequencies in a wide range (100-1000 MHz). In each tested frequency, we turn on the EMI signals for about 2 seconds and then turn them off. The 2-second duration corresponds to the 2-second press event many smart glasses are designed to respond to. We generate continuous single-tone EM signals using a directional antenna [61], a MiniCircuit ZHL-20W-13+ amplifier with a maximum EMI transmission power of 20 W [62], and an Agilent N5172B vector signal generator. As shown in Fig. 6, we observe that injections of EMI signal at specific frequencies can produce a touch effect on smart glasses to activate voice assistants. We are able to attack the Eyeweare Plus smart glasses with EMI signals at 387 MHz at an 8.0 $cm$ distance to invoke the VA (Fig. 6). Following the same process, we test the Razer Anzu smart glasses, and can activate the VA with EMI signals at 358 MHz at a 1.2 $cm$ attack distance. The attack is effective in activating smartphone VA systems when the smart glasses are paired with Android or iOS phones in lock-screen status.

To further understand the potential causality, we tear down the Razer Anzu and the Eyewear Plus smart glasses. The internal structure shows the electrodes of capacitive touch sensors inside the touch area on smart glasses' frames (Fig. 13, see Appendix A). Based on these observations and the principle of capacitive touch sensing [63], [64], [65], we build a prototype touch-sensing circuit (Fig. 14, see Appendix B), including a resistor, a microcontroller (Arduino), and an electrode (5 × 7 $cm^2$ conductive copper surface). As illustrated in Fig. 7, the microcontroller's output pin repetitively charges and discharges the circuit. When a conductive object, such as a finger, touches the electrode, the capacitance value will increase. The RC time constant [66] $t = RC$ represents the charge time of the circuit, where $R$ refers to the resistance value in the circuit, and $C$ is the capacitance value. When the capacitance increases,

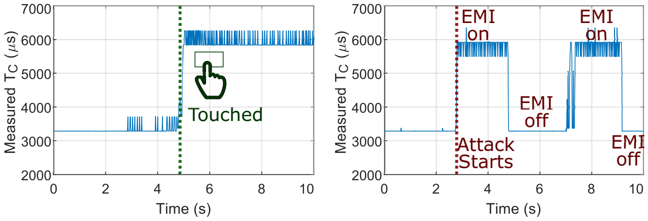


Figure 8: Measured charge time ($T_C$) with touch (left) and EMI (right).

the charge time will also be extended. By monitoring the input voltage $V_{in}$, the microcontroller measures how long it takes to charge the circuit to the set voltage threshold ($V_{th}$) and utilizes the measured time ($T_c$) to detect touch events (Fig. 7). Fig. 8 shows the measured charging time ($T_c$) with touch or specific EMI signals. We can observe that EMI can also induce changes in $T_c$, which could lead to false detection of the touch. In this experiment, we generate EMI signals for about 2s and turn off EMI signals. We then generate EMI signals for another 2 seconds. By monitoring the input voltage signal, we observe that EMI signals can induce noises and DC offsets in the signal of $V_{in}$. Thus, our hypothesis is that the injected EMI signals affected the input voltage $V_{in}$ and subsequently altered the measured change time $T_C$, which in turn affects the perceived capacitance.

Since the actual touch sensing circuit in smart glasses is in a black box, we use the prototype circuit in this preliminary analysis. This prototype circuit is based on the capacitive sensing principle [63], [64], [65] but may not fully represent the actual touch sensors being used in smart devices. For instance, the actual implementation of capacitive touch sensing in devices may use complex schemes in charging/discharging the circuit and measuring the average charging time [67], [68]. We also notice that the devices usually have a self-calibration mechanism that adjusts the reference measurement every several seconds when there is no touch [69]. Depending on the implementation of real-world devices, the attack effects can be caused by a combination of effects that alter the voltage signal in analog sensor components [29], [54], [55], [56] or induce changes in touch sensing controllers [49], [52], [53].

**Attack Smart Glasses with Oscillating Circuits.** After verifying the effects of EMI on smart glasses, we investigate the feasibility of increasing the effectiveness of the attack using low-cost devices.

Specifically, we experiment with circuits based on 3-point capacitor oscillating circuits (Colpitts oscillator) [70]. The Colpitts oscillator takes feedback from a voltage divider made of two capacitors in series across the inductor. The circuit is a kind of LC oscillator, using a combination of inductors (L) and capacitors (C) to produce an oscillation at a certain frequency. The circuit we use consists of a bipolar junction transistor (NPN transistor C2078), which is used as the gain device. Compared to RF-generating equipment that can be expensive and heavy, oscillating circuits can be purchased at a low price [71], [72] and powered by a simple DC power supply or batteries. Furthermore, oscillating circuits could generate EMI signals of higher intensity, which could increase the effective attack distance.

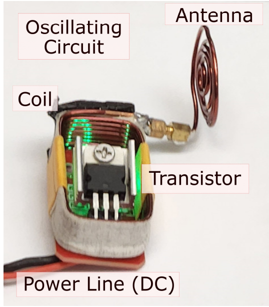
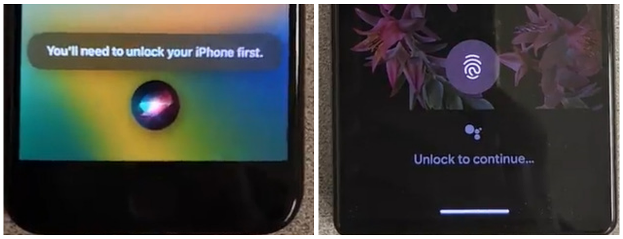


Figure 9: The oscillating-circuit-based attack device consists of an NPN transistor, coils, and an antenna.

We power the oscillating circuit with a DC power supply and connect the circuit with a small spiral antenna (Fig. 9). By using the oscillating circuits, we are able to achieve a 20.6 $cm$ attack distance to activate VA systems using EMI on the Eyewere Plus glasses and 12.2 $cm$ on the Razer Anzu smart glasses. In comparison to attacks at 8 $cm$ and 1.2 $cm$ using typical RF antennas and equipment in Section 4.1, the attack distance has been substantially extended. We use an Agilent MSO-X 4054A oscilloscope to measure the frequency of the oscillating circuit. The input power of the oscillating circuit is around 28.00 V $\times$ 0.95 A = 26.6 W. The oscillating circuits we experiment with usually have a specific frequency within the range of 40 - 50 MHz. In real-world attacks, adversaries can try to find the attack frequency using wide-range RF devices to find suitable attack frequencies by sweeping. The adversaries can also modify/build attack circuits [70] using different circuit parameters to adjust the attack frequency. The power can usually be controlled by adjusting the amount of input DC voltage of the oscillating circuit. It can be adjusted based on the distance following the inverse-square law to achieve specific electromagnetic field intensity.

## 4.2. Bypassing Verification of Application Initiation

**Voice Interaction with Locked Phones.** We first investigate whether controlling the Tesla app via normal interactions with smartphones' VA requires unlocking the phones. ‘For iOS devices, we use Semantically - similar voice commands to test if Siri can invoke Tesla app and realize the functions included in the commands, such as “unlock my Tesla,” “unlock car, “use Tesla app to unlock my car,” etc. We observe that although Siri is aware of the intention of the users, *it asks the user to unlock the phone before executing the Tesla functionalities*. For Android devices, Google Assistant does not directly connect to the Tesla app. We thus ask Google Assistant to open the Tesla app, but it also requires the user to unlock the phone first (Fig. 10). Our results thus show that adversaries cannot perform malicious control over the connected cars without somehow unlocking the


You'll need to unlock your iPhone first.

Unlock to continue...

Figure 10: When injecting voice commands without using automation tools, the iOS (left) or Android (right) systems request unlocking the phone first to invoke the Tesla app.

victims' phones, which is challenging as they are protected by password or biometric authentications.

**Unlocking-free Trigger Phrase Injection.** Surprisingly, our experiments show that directly injecting trigger phrases into automation tools through the VA interfaces allows adversaries to bypass the step of unlocking phones. Furthermore, the target car control apps do not even need to remain open before the attack.

**Exploiting iOS Shortcuts.** As mentioned above, if we use "Hey, Siri. Unlock my Tesla", the system will only perform the functionality after users unlock their phone first. However, if we play the audio of "Unlock Tesla", which is the *default prompt in Shortcuts for unlocking Tesla, the corresponding task will be executed without asking to unlock the phone*. We test with other commands related to Tesla in Shortcuts (Fig. 4) and verify that the default Tesla commands can all be executed without unlocking the phone.

The causality of the difference between the above two operations can be attributed to the design of Shortcuts. According to Apple [73], when providing Shortcuts, the acceleration that the app offers should be substantial - the app designer should not just expose a shortcut that does about the same thing as opening the app normally [73]. The app's functionality is exposed to system services, such as Shortcuts app, by implementing the AppIntent protocol. The protocol provides trigger phrases of the function, the needed data for the function, and the codes to perform the function [74]. The system will then expose the actions directly from the Shortcuts app and indirectly through natural language commands spoken to Siri [74]. When a user invokes a function through Shortcuts, the system instantiates an app intent using the *init()* initializer. The system sets parameters based on user input or other available sources and calls the *perform()* function to perform the app intent [74]. Starting with iOS 15 and macOS 13, an app designer can create a preconfigured App Shortcut that allows users to discover and run the app intent without any configuration [75]. By creating App Shortcuts, the app's functionality becomes instantly available for use in Shortcuts and Siri from the moment a user installs the app, without any setup in the Shortcuts app or an Add to Siri button [75].

**Exploiting IFTTT.** We experiment with IFTTT setup suggested by Tessie [16] on Android. Similar to the iPhone, we verify that there is no need to unlock the phone's screen to control security-sensitive functions. We can unlock the Tesla's doors, open the Tesla's front/rear trunks, remotely start the car, etc.

**Summary.** As suggested by Apple, the app designer exposes only shortcuts that are executable at any time, without relying on the user being in some particular state before the shortcut will be ready for use. When the user wants to use the Shortcuts in Siri or on the lock screen, the app or app extension will be ready to be invoked and be handed the shortcut to handle [73].

While these design choices provided by platforms and device manufacturers [1], [6] greatly accelerate performing tasks, they can introduce vulnerabilities into the connected systems. We identify the following key weaknesses: 1) Although directly asking the VA system to invoke functionalities of the Tesla app requires unlocking the phone first, there is no need to unlock the phones when invoking functionalities via automation tools. 2) The voice commands are available for anyone to use. The default voice commands can be found from online resources [6], [7], [16], [76]. Moreover, our evaluations show the system accepts voice trigger phrases with different accents and gender (Section 5.3). 3) There is no need to open apps (such as Tesla or Tessie app) or automation tools in the background during the attacks.

Overall, we have demonstrated utilizing malicious physical signals to non-invasively exploit the automated control chains with Tesla. The attack targets the vulnerabilities in the intersection of smart wearable and mobile devices and automation frameworks. We observe that the devices completely trust and accept physical signals, including malicious EMI and synthesized voices, to execute automated tasks. Moreover, the adversary does not need physical access to the victim's devices nor compromising conventional security mechanisms, such as account credentials and password/biometric-enforced screen locks.

## 5. Evaluation

In this section, we evaluate the attacks in different combinations of smart glasses, smartphones, and automation platforms like Shortcuts and IFTTT. Additionally, we evaluate the effects of noise and commands in different voices. We evaluate our attacks in indoor and outdoor scenarios.

### 5.1. Experiment Setup

We validate the effects of the proposed attack on a 2018 Tesla Model 3 and a 2023 Tesla Model Y. The vehicles are in their default configuration, which allows keyless driving after remote start. We first perform the attack via the Tesla app on iOS devices. Once the Tesla app is installed, the Tesla Shortcuts will be automatically installed in the system. Later the corresponding widget will appear in the Shortcuts app. We then investigate the attacks on Android devices that have installed IFTTT and configured IFTTT applets based on the Tessie documentation [16].

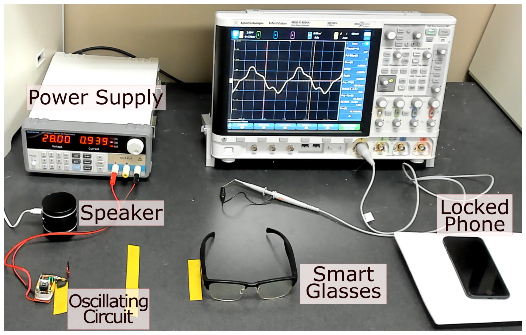


Figure 11: Experimental settings of physical signal injection attacks on smart glasses with the oscillating circuit and a speaker. The phone is in lock-screen status during the attack.

Fig. 11 shows the indoor experiment settings. We power the oscillating circuit with a DC power supply. We play voice commands with a compact speaker that has a size of 1.9 × 1.9 × 1.6 inches and 3W RMS power. We describe the outdoor experiment settings in Section 5.4.

## 5.2. Attack Evaluation on Smart Glasses

We pair the smart glasses with an iOS (iPhone SE, 3rd Gen) and an Android phone (Pixel 6) respectively. We then conduct end-to-end attacks on the smart glasses to manipulate Tesla vehicles. Table 1 summarizes the security-sensitive commands we have tested. Specifically, we test functionalities such as unlocking the car and opening the front/rear trunks. Additionally, we test disabling the sentry mode that records and reports suspicious activities near the vehicles. We also test manipulating the charging port or stopping charging the vehicle using physical signal-based attacks. Further, we test executing B7 “activate remote start” after executing B1 “activate unlock” by physical signal-based attacks to enable keyless driving, and we verify that the adversary will be able to drive the car away.

Our experiments show that the functionalities (A1-A8, and B1-B9) can be executed by attacking smart glasses without requiring user verification and interaction. Only when playing A5 and A6 to open the trunks with Apple Shortcuts, the VA system in the iPhone ask for confirmation over the glasses. However, after playing the sound of “sure” or “yes’, the system will proceed to open the rear or front trunk. The only command that often fails to execute is “Enable Dog Mode on Tesla”. However, this command is difficult to execute even when the user uses Siri and Shortcuts normally, which might be because of the pronunciation clarity of the speech. We observe that when we play the AI audio clips with UK accents [77], the system can recognize the phrase slightly better.

In addition, we evaluate our attacks on five smart glasses from four manufacturers. We find that the maximum distance to activate the VA system without physically touching the smart glasses is above 10.2 $cm$. For certain smart glasses, the distance can reach over 20 $cm$.

TABLE 1: Tested Shortcuts and IFTTT Applets on screen-locked phones

| Shortcuts | | IFTTT Applets | |
|---|---|---|---|
| A1 | Unlock Tesla | B1 | Activate Unlock |
| A2 | Lock Tesla | B2 | Activate Lock |
| A3 | Open Tesla Charge Port | B3 | Activate Front Trunk |
| A4 | Close Tesla Charge Port | B4 | Activate Rear Trunk |
| A5 | Open Tesla Frunk | B5 | Activate Vent Windows |
| A6 | Open Tesla Trunk | B6 | Activate Close Windows |
| A7 | Disable Sentry Mode on Tesla | B7 | Activate Remote Start |
| A8 | Stop Charging Tesla | B8 | Activate Open Charge Port |
| A9 | Enable Dog Mode on Tesla | B9 | Activate Close Charge Port |

As shown in Table 2, we evaluate the attacks with two Shortcuts and two IFTTT commands and execute each command ten times to observe the success rate and average attack time. We play the voice synthesized by NaturalReaders [77], an online Text-to-Speech (TTS) platform. We use an English male voice with a UK accent (Oliver). The environmental noise in the lab is about 47.6 dBA. We find that the attack is almost always successful. The average time is about 5.5 seconds to 9.3 seconds for short commands such as A1 (unlock Tesla) and B1 (activate unlock). While for longer commands A7 and B5, the attack can take longer time.

We observe that the Vue smart glasses are connected to smartphones via an app, which seems to degrade the communication speed and introduces delays to the attack process. The attack time for SD-G3 smart glasses is also longer because it requires a 3-second press while other smart glasses require 2 seconds. Thus, the EMI activation phase lasts at least 3 seconds for SD-G3 smart glasses.

## 5.3. Attack Evaluation with Different Voices and Control Chain Modules

**Voices.** We use artificial voice clips that vary in gender and accent as action commands. As shown in Table 3, we test four voices in both Siri and Google Assistant, including male and female, with UK or US accents. The voice names are Oliver (UK Male), Bella (UK Female), Guy (US Male), Jane (US Female), respectively. The background noise is around 47.6 dBA during the experiments.

We play the above synthetic voice clips to the microphone of Razer Anzu smart glasses paired with different phones. For each combination of voice and VA platform, we test 10 times for each command in Table 3 and record the success rate: A1 and A7 for Siri; B1 and B7 for Google Assistant. We first test an iPhone SE (3rd Gen) owned by a female user, with Apple Siri (iOS 16.6) on it. We find that the attack accuracy is high regardless of voice sources. We then test a Pixel 6 owned by a male user, with Google Assistant (Android 13). We find that the attack success rate is 100% for all four attack voices. Our observations validate that the VA systems do not verify the users’ voices in our attacks.

**Control Chains.** We validate our attacks on different control chain modules. Table 4 lists the operation system versions, phones, subsequent control chain modules, and the validated

TABLE 2: End-to-end attack results on smart glasses to Tesla via Shortcuts and IFTTT Applets on locked phones

| Smart Glasses | Max Activation Distance ($cm$) | Tested Shortcuts/ IFTTT Applets | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | A1 | | A7 | | B1 | | B5 | |
| | | Succ. Rate | Aver. Time | Succ. Rate | Aver. Time | Succ. Rate | Aver. Time | Succ. Rate | Aver. Time |
| Razer Anzu | 12.2 | 10/10 | 8.0 s | 10/10 | 7.2 s | 10/10 | 7.3 s | 10/10 | 7.0 s |
| Eyewear Plus | 20.6 | 10/10 | 5.5 s | 10/10 | 8.3 s | 10/10 | 8.0 s | 10/10 | 9.0 s |
| Eyewear Pro | 20.2 | 10/10 | 6.0 s | 10/10 | 8.8 s | 10/10 | 6.2 s | 10/10 | 7.3 s |
| Vue | 13.0 | 10/10 | 7.5 s | 9/10 | 10.0 s | 10/10 | 6.8 s | 9/10 | 10.4 s |
| SD-G3 | 10.2 | 10/10 | 9.3 s | 10/10 | 10.0 s | 10/10 | 8.6 s | 10/10 | 8.8 s |

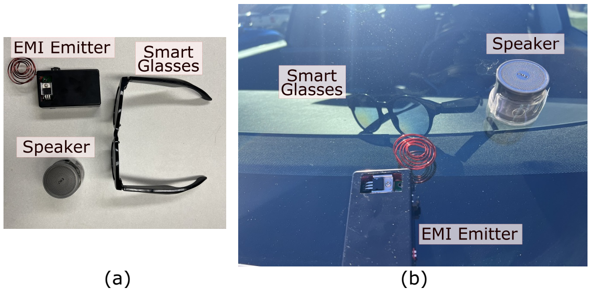
TABLE 3: Results of evaluation with different voice accents, genders, and different VAs.

| Attack Voice | VA Platform and Owner Gender | | Success Rate | |
|---|---|---|---|---|
| | Platform | Gender | A1/B1 | A7/B7 |
| UK Male | Apple Siri | Female | 9/10 | 10/10 |
| UK Female | | | 10/10 | 10/10 |
| US Male | | | 10/10 | 9/10 |
| US Female | | | 8/10 | 10/10 |
| UK Male | Google Assistant | Male | 10/10 | 10/10 |
| UK Female | | | 10/10 | 10/10 |
| US Male | | | 10/10 | 10/10 |
| US Female | | | 10/10 | 10/10 |

commands. In iOS devices, the system invokes codes in the Tesla app via its exposed Shortcuts and then communicates with the Tesla API to control the car. In Android devices, the Google Assistant accesses the IFTTT applets to send a Webhook request to Tessie API. The Tessie server then communicates with Tesla API to control the car. We verify that the commands A1-A8 (B1-B9) can be successfully executed via the attacks in these configurations.

We further configure a longer customized control chain. In this control chain, we use Apple Shortcuts to trigger another automation tool (IFTTT) to access the server APIs, which is the longest control chain shown in Table 4. Specifically, we configure the IFTTT applets to be invoked by Apple Shortcuts and customize the Shortcuts to trigger the IFTTT applets. The IFTTT applets then send Webhook requests to the Tessie API, which communicates with the Tesla API to control the car functionalities. We list the customized Shortcuts command in Table 7 (see Appendix B) and the control chain modules as well as the validated commands in Table 4. The results validate the attack feasibility on customized control chains, indicating potential risks on more diverse and complex control chains as automation tools become increasingly powerful and interconnected in the future, as users share their Shortcuts and IFTTT applets in the community, and as more 3rd-party providers offer services via the automation tools.

The results validate our key observation on the threats lying in the wearable technology and automation systems: once modules in automated control chains are installed and configured, the system may not effectively verify the user information even when the control chains are triggered by adversaries using physical signals that are independent of the user.



(a) (b)

Figure 12: Devices to attack smart glasses in a Tesla car.

### 5.4. Attack Evaluation in Different Outdoor Scenarios and Noise Levels

**Outdoor Scenarios With Ambient Noises.** We choose four scenarios (Fig. 15, see Appendix C) for real-world tests: a parking lot in front of a coffee shop with a background noise level of 56 ∼ 69 dBA, a supercharger station parking lot with a background noise level of 63 ∼ 66 dBA, the parking lot in front of a supermarket with a background noise level of 57 ∼ 61 dBA, and a relatively quiet university parking lot with a background noise level from 49 to 51 dBA. For each scenario, we verify the effectiveness of the attack on a Razer Anzu and an Eyewear Plus smart glasses and inject each voice command (A1-A8) and (B1-B8).

We then inject specific commands and evaluate the success rate to explore the effects of noises. We test "unlock Tesla" 10 times by attacking Eyewear Plus smart glasses paired with an iPhone in each scenario. During the attacks, the victim's smart glasses are placed on the console inside the car, as shown in Fig. 12b. Table 5 summarizes the attack success rates in different scenarios.

**Outside Scenario with Controlled Noises.** We now evaluate the attack performance under different controlled noise levels. During the experiments, we place two full-range speakers in front of the car to generate noise. We use white noise generated by Audacity and publicly available high-fidelity recordings of human conversation and traffic noise [78], [79]. We gradually increase the sound level of white noise and observe the changes in the attack success rate. We choose the noise level starting from 70 dBA based on two reasons: 1) we try to evaluate the attack result based on higher-level noise; 2) the mean urban street noise level is 73.4 dBA, with substantial spatial variation (55.8-95.0 dBA), from the statistics obtained at 99 street sites throughout New

TABLE 4: Validates attacks against different combinations of the automated control chain modules.

| Smart Glasses Models | Smart Phone Models | OS | Voice Assistants | Subsequent Automated Control Chains | Commands |
|---|---|---|---|---|---|
| Eyewear Plus | iPhone SE (3rd Gen) | iOS 16.6 | Apple Siri | Shortcuts → Tesla app → Tesla API | A1-A8 |
| Razer Anzu | iPhone 12 | iOS 16.3 | Apple Siri | | |
| SD-G3 | iPhone 13 Pro | iOS 16.6 | Apple Siri | | |
| Eyewear Plus | Samsung Galaxy S10 | Android 11 | Google Assistant | IFTTT → Tessie API → Tesla API | B1-B9 |
| Razer Anzu | Google Pixel 6 | Android 13 | Google Assistant | | |
| Eyewear Plus | iPhone 12 | iOS 16.3 | Apple Siri | Shortcuts → IFTTT → Tessie API → Tesla API | C1-C9 |
| Razer Anzu | iPhone SE (3rd Gen) | iOS 16.6 | Apple Siri | | |

TABLE 5: Noise and success rate of (A1: unlock Tesla) under different scenarios.

| Scenario | Ambient Noise (dBA) | Success Rate |
|---|---|---|
| Coffee shop parking lot | 56 ∼ 69 | 10/10 |
| Supercharger station | 63 ∼ 66 | 10/10 |
| Supermarket parking lot | 57 ∼ 61 | 10/10 |
| University parking lot | 49 ∼ 51 | 10/10 |

TABLE 6: Controlled Noise and success rate

| Noise | Sound Level (dBA) | Success Rate |
|---|---|---|
| White noise | 70 | 9/10 |
| | 75 | 8/10 |
| | 80 | 6/10 |
| | 85 | 5/10 |
| | 90 | 6/10 |
| Conversation | 67.3 ∼ 84.4 | 9/10 |
| Traffic | 77.6 ∼ 87.6 | 10/10 |

York City in 2015 [80]. Sound level ranges are recorded for the noise because the sound level fluctuates as the audio plays. We summarize the results in Table 6.

Two attack devices are used in outdoor attacks: a portable electromagnetic signal generator based on an oscillating circuit and an EWA A109 mini speaker. Fig. 12 illustrates the devices and outdoor attack settings. The portable EMI device is based on the same oscillating circuit for indoor experiments but is powered by batteries. They cost about $28 [81] and $19 [82], respectively, and can be purchased from Amazon and Walmart. The total cost of the portable EMI device and the mini speaker used in such an attack is around $47.

# 6. Discussion

## 6.1. Potential Impact Analysis

We have demonstrated it is possible to launch the proposed attacks. Next, we discuss the possible relatively severe consequences if our attack is successfully conducted by people with bad intentions. After unlocking the vehicle, in addition to stealing the victim's belongings, they can install malicious devices inside the car. Such malicious devices range from malicious OBD dongles to eavesdropping, location tracking, and video surveillance devices. Furthermore, they can hide the contraband in the victim's car and transport it through the car. They can even perform some operations that threaten driver safety, such as physically breaking the steering wheel and braking system, robbery, and kidnapping. By "remote start" command injection, the target vehicle can be driven away directly. Although Tesla can be traced by the smartphone app, the high demand for Tesla cars still is the stimulus of the motivation behind the theft. For instance, German police found a dismantled Tesla Model S inside a truck near Germany in 2017, which was discovered as the one stolen in the Netherlands four days ago. The thieves dismantled the vehicle to evade app detection [83].

## 6.2. Countermeasures

Completely addressing risks in the control chain can be difficult due to the involvement of multiple vendors and the integration of their upstream and downstream products that have already been trusted. Each of them hopes to be compatible with more devices and provide more functions to attract consumers. However, more elements'integration means more potential risks beneath the connection. Almost every day, products from third-party suppliers or vendors appear to join a communication chain and can be potential targets. Any of these is likely to have the weakest cybersecurity. Such chained hazards are hard to predict and cannot be solved without the cooperation among related vendors. We try to decompose the attack process and find countermeasures for the main sublinks to mitigate this risk. We also attempt to propose a prevention suggestion to mitigate the effects caused by attacks on connected devices.

**6.2.1. Requiring to Unlock Phone for security-critical automation.** Simple fixes like additional authentication for critical apps could alleviate the threats. However, extra steps added to routine activities like unlocking/locking cars could compromise the convenience and accessibility of automated systems. If the users are required to frequently unlock their phone's screen, the benefits of automation tools could be sacrificed. Similarly, a user could disable voice assistants in lock-screen statuses, which will affect the usability of voice-associated functionalities. In the future, it might be possible to provide more fine-grained authentication methods by tailoring continuous authentication methods [84] and event-based approaches [85] in the context of automated control chains. Further, it might be possible to use anomaly detection algorithms to determine the likelihood of attacks and ask the user to perform additional authentication methods only if an anomaly is detected. For example, when the user's location is far away from the vehicle, an unlocking or remote start command may require further authentication, such as asking the user to unlock the screen.

### 6.2.2. Defense against Voice Spoofing Attack.

**Multi-modality Verification.** As a passive verification mechinism, human feature detection modules can be added to verify the usage status of smart glasses. They include skin detection sensors, microphones can detect airflow changes triggered by popping sound [86] and oral cavity movement [87], [88], [89], [90], [91]. Some smart glasses [92] have already embedded IMU sensors to capture their status. They can also be used for liveness detection when accepting voice commands. A simpler solution might be integrating an infrared module into the frame to detect if there is a barrier between two temples to detect whether glasses are on users' heads. However, adding extra modules will increase the cost and influence the portability of smart glasses.

**AI Distinction.**Existing researches explore intrinsic differences in characteristics and coefficients between bonafide voices and synthetic voices from Text-to-Speech tools to build models for classification. They include vocal tract, voice textures, Gammatone Cepstral Coefficients and Mel-Frequency Cepstral Coefficients [93], [94]. Loudspeakers are also found always introducing distortions to the sound they generate and circuit noise. AI algorithms could distinguish between real voice and voice from speaker [89]. These methods introduce extra computation overhead and may increase the response time.

**Customization of Voice Commands.** We considered renaming the related commands in Shortcuts, such as changing "Unlock Tesla" to "Unlock." We assumed that if the modified commands are different from the injected default ones, the corresponding commands cannot be invoked. We observe that: 1) the changed commands for IFTTT applets can be effectively against such attack, and 2) we also can change the Tesla-related commands in iPhone Shortcuts, but the corresponding default commands are still available even after modifications. This means that the adversaries can still attack the system utilizing the default voice command phrases. Further, the adversaries may guess the commonly used commands and try each of the commands in the attack.

**6.2.3. Defense against EMI Attack.** More robust electromagnetic shielding may protect sensors from EMI. However, it may be challenging to completely shield a capacitive touch sensor that usually requires non-conductive external material around the touch area. Researchers [60], [95] designed specific sensor defense methods against EMI. Zhang and Rasmussen [95] proposed to detect EMI signals by using a high-speed switch to turn the circuit on and off based on a secret sequence. Since capacitive touch sensors are operated differently, a switch may affect the detected charge time of the circuits. Nevertheless, we could detect non-zero samples caused by EMI when the circuit has been discharged and the input voltage reading should be 0. Likewise, if the input voltage is low after the circuit has been charged for an extended time. The reading is not likely to be true because it may exceed a normal threshold of the device's typical use cases. Further, EMI can cause irregular patterns of the detected voltage and the measured capacitance value. Future research might explore utilizing machine learning methods to distinguish the patterns resulting from EMI and actual touch events. This approach may lead to an increase in the power consumption.

**6.2.4. Risk Mitigation Proposal on Control Chain.** Weak links in control chains are easier targets for cybercriminals. As the pivot point of the entire control chain, smartphone system and app designers are appealed to be more aware of the security implemented within each step of their control chains and have stricter and more comprehensive test standards to add new functions and connect new devices in their chain. For third parties, their potential risks should be evaluated. The phone company can ask vendors to perform self-assessments and prove that they are secure vendors to join the chain.

The over-privilege of the third-party automation tool is another issue that should be addressed. Vendors expose immense amounts of data and considerable control authority to the automation platforms, which enlarges the attack surface. Once the automation platforms are hackable [96], attackers find a "side door" to access the components connected to the automation platforms. For example, in the attack we proposed, by hacking these tools, the attackers are able to gain control over all Teslas connected to these platforms.

## 6.3. Limitation & Future Work

**Attack Distance.** The attack distance with our experimental setting is limited to about 10 - 20 cm. In real-world scenarios, determined resourceful attackers can employ higher-end devices such as high-power EM generators and directional antennas to increase the distance, as has been shown or discussed in previous works [52], [55], [97]. While the attack range could be extended, it is yet to be investigated whether the attack remains low-cost when using a more advanced attack setup to increase the attack distance. Since the electromagnetic field strength decreases as the distance increases, the adversary could adjust the power based on the required distance to achieve consistent attack effects. Longer attack distances can be achieved with a more sophisticated setup (e.g., using directional antennas, enhancing heat dissipation, and incorporating an array of emitters and higher-power devices) [98], [99], [100].

**Attack Stealthiness.** We demonstrated that with a 3W speaker and lower audio output volumes, the attack is still effective even on sites adjacent to streets that have heavy traffic flow. However, it is still possible that the attacks can be noticed by nearby pedestrians. Future works may investigate the feasibility of utilizing inaudible carriers [17], [19] to inject signals into the microphones of smart glasses. For example, we are able to inject audio signals into the microphone of Razer smart glasses using laser. However, we notice that it may be challenging to inject laser into certain devices such as Eyewear Plus smart glasses because the internal microphone does not directly face the hole in the plastic surface of the device. In this paper, we mainly consider proof-of-concept attack implementations utilizing devices that can be purchased at a low cost and readily used by the attacker in both indoor and outdoor settings.

**Attack Scenarios.** In our attack setting, we assume that the victim does not wear smart glasses during the attack. Otherwise, the victim may hear the response of the voice assistants and become aware of the attack. Future works may investigate the feasibility of exploiting inaudible channels to tune down the volume of the victim system and inject voice signals, as shown in [19], [20], [101].

**Beyond Touch Sensors.** Although touch sensor-based activation is widely used by smart glasses, our work did not explore other potential designs of activation methods exhaustively. As a result, the specific attack presented in this paper may not work on certain smart glasses that do not use touch sensors. However, the underlying principle of EMI-based automation chain takeover applies to other similar wearable smart devices. We believe future work can investigate the physical vulnerabilities of other types of sensing structures based on our methodology.

**More Control Chains.** Future research should extend to explore security issues in control chains involving a broader range of connected vehicles, wearable devices (such as the trending smart ring), mobile devices, and smart home systems, with the goal of fostering a more secure digital environment in an era of ever-increasing connectivity. For example, since Tessie's interfaces are available via Webhooks, they can potentially open up for a wide range of attacks in a similar fashion to IFTTT. In a worse case, if the third parties (IFTTT and Tessie) are compromised, the attack surface will be significantly increased [96].

Besides Tesla, we have already experimented with a customized control chain on a customized car (2015 Chrysler 200). Specifically, the control chain involves smart glasses, iPhone Siri, iOS Shortcuts, the Shortery app [3] that activates macOS Shortcuts [102], which run a Node.js script to communicate with Drone Mobile API [103]. It then controls the Chrysler car, which is installed with a Compustar [104] vehicle security system and a Drone Mobile system that allows software-based access to unlock/lock and remotely start the car. We configure the customized control chains in iPhones with commands: "Unlock my car" and "Lock my car". We are also able to manipulate the car's functionality by attacking the smart glasses paired with the iPhones via the customized automated control chain.

Furthermore, similar attacks could be generalized to other automation with voice shortcuts, as VA penetrates more into consumer markets, such as smart-home (doors, garage door), industrial production, medical robot (surgical robot, robot for precision drug delivery).

### 6.4. Ethics and Responsible Disclosure

All experiments were conducted on our own devices and vehicles, ensuring no unauthorized testing on external or third-party systems. Further, we have taken proactive steps to disclose our findings responsibly to relevant stakeholders. This includes reaching out to manufacturers of the smart glasses used in our study, as well as major technology companies such as Apple and Google, and automotive company Tesla. These disclosures were made with the intent to contribute to enhancing security measures in their respective products and systems, thereby fostering a safer technological environment for all users.

For the consideration of health issues that might be introduced by radiation, we have measured the electromagnetic (EM) and electric (E) fields around the attack device with an EMF meter. At 10cm, the maximum measured EM and E fields are 6.5mG and 35V/M, respectively. At 30cm, we did not detect any disturbances in the fields resulting from the attack, indicating that the fields in our experiments are unlikely to cause critical concerns as they are comparable to small household appliances and below the safety limits [105].

## 7. Related Work

**Attacks on Voice Assistant Systems.** The previous work can be categorized into two groups depending on whether the adversaries need to access the victim's device. One group requires neither physical contact nor software change of the device; in contrast, they simply transmit voice signals to the microphones of victim devices using a speaker or other injection devices that carry out silent voice command injection. For the audible cases, they hijack other nearby devices to play malicious voice commands [106] or play adversarial samples of legitimate voice commands generated by using machine learning algorithms [86], [107], [108], [109], [110], [111], [112], [113]. The voice commands that can be analyzed by machine are hidden in the sound that is recognized as normal noise or music by humans. For the inaudible remote attacks, the attackers either directly inject unwanted voice commands by electromagnetic interference [29], [114], [115], [116] or utilize other inaudible mechanical waves to load the injected voice command [17], [18], [19], [20], [48], [101], [117], [118], [119], [120], [121]. Those inaudible carriers usually have high frequencies beyond 22kHz, such as ultrasound and laser. The other group requires altering the software or physically contacting the victim's device. For instance, Google Voice Search is the first hacked VA system; the adversary directly manipulates the victim phone's speaker to play malicious voice commands by triggering the preinstalled malware on the Android phone [122]. Young et al. connected the victim's phone and the attack module (consisting of a Raspberry Pi and an audio card) with a special audio line to spoof the victim's phone to recognize it as a paired headphone. The tool has the ability to simulate the 'middle button press' to activate VA

[3] Shortery allows running Shortcuts automatically on macOS computers https://apps.apple.com/us/app/shortery/id1594183810.

and then play the recorded voice commands [123]. Wang et al. injected malicious voice signals with conducted EMI by modifying a charging device plugged into the victim's phone [124].

In comparison, our attack does not require physically contacting the victim's device to implant malicious malware, plug any wires, or connect any peripheral device to it. This presents a more practical attack with low-cost attack devices that can be easily purchased from stores. Most importantly, in manipulating vehicle apps that are considered security-critical, we could bypass PIN/biometric authentication. We also do not have the constraints of collecting victims' voice samples and building machine-learning models to produce adversarial voice clips.

**EMI Injection Attacks.** Prior studies [29], [49], [52], [54], [55], [58], [59], [97], [125], [126], [127] have shown that far-field and near-field electromagnetic (EM) signals can manipulate sensors, touchscreens, keyboards, and actuators. Especially, existing works showed how intentional EMI can affect a victim's smartphone touchscreen via a malicious table [49], [52], [53] or a modified charging cable [50], [128], [129]. Since these attacks directly target smartphone screens to induce events via the graphical interface, the attack approach would still require unlocking the phone's screens to manipulate the app's functionalities, which usually need additional social engineering efforts. Our study adopts a different attack path and targets the capacitive touch sensor on a connected device (smart glasses) that is different from the smartphone itself and explores the attack effects in the context of control chains while the phone is in a lock-screen status. The novelty of our study lies in: 1) discovering the impact of exploiting smart glasses as an entry point for the first time, 2) achieving unauthorized automated control chains (circumventing conventional requirements of unlocking screens [19], [49], [50], [53], [129] nor requiring the owner's voiceprint to activate VA [19], [20], [101], [116], 3) characterizing the threats and inherent weaknesses posed by the control chains, and 4) demonstrating a non-trivial end-to-end attack to compromise modern Tesla vehicles. To realize the end-to-end attack, our approach reveals systematic unintended/insecure behaviors in connected components, including official Apple and Tesla hardware/software/interfaces that have undergone continuous, rigorous security analyses. We explore and validate flaws in automation framework designs relying on blind **transitive** trust. Without wider community awareness, these flaws are bound to persist across numerous critical systems, resulting in concealed risks as more systems become interconnected.

**Attacks on Automotives' Remote Controls.** Passive Keyless Entry and Start (PKES) system detects the proximity of authorized mobile devices or key fob based on signal strength (RSSI) and latency measurements of cryptographic challenge-response operations conducted over BLE [130]. Once within a range, users can operate the vehicle without interaction with the above keys. The attackers use a replay attack to spoof the distance detection system by capturing the radio signal sent from the owner's key fob to the car and playing it back later. The signal is designed to match the code saved in the car to verify ownership. Replay attack is applied even for the modified rolling code system since researchers found that the counter at the vehicle end will be re-synced and commands from the previous cycle of the counter will work again [130], [131], [132], [133], [134], [135], [136], [137], [138], [139]. One research sniffs and analyzes the signal first, and then injects noise to mislead the receiver to reduce its measured distance [140]. The hackers also exploits the NFC' vulnerability of Tesla's keycard to unlock or drive it away [141].

The CAN Bus data or firmware can be modified for attacks [142], [143], [144], [145], [146], [147]. The CAN bus has no authentication or encryption schemes. Through reverse engineering, once hackers discover which certain behavior a specific CAN frame can trigger, they can send specific CAN frames with the right CAN ID and data payload into the target CAN bus. Keen Security Lab of Tencent also explored the browser vulnerability of Tesla Model S/X to communicating with Electronic Control Units (ECUs) on the CAN bus [142], [143]. The attacker faked the Android Over The Air (OTA) firmware, embraced the unauthorized remote unlock command, and distributed it. The victim telematics device on the vehicle downloaded and installed the modified firmware. The attacker then took some control of the vehicle [145].

Telematics units can be compromised by installing malicious code through a wired or wireless connection [133], [148]. CD, OBD2 port, PassThru, and Bluetooth/Cellular, can all be utilized as the path to install code into a vehicle for disabling its security measures and compromise telematics unit to unlock the doors or control car charging remotely [133], [148], [149]. Researchers also executed an attack on Tesla's Gateway energy management system to open the frunk or door of a Tesla Model 3 even though the car was in motion [150].

Corresponding control app is another attack surface [137]. Researchers exploited vulnerabilities in mobile apps in Hyundai and Genesis car models after 2012. They found that validation of the owner is done based on the user's email address, included in the JSON body of POST requests. They created another account using the target's email address followed by a control character to bypass the validity check in Hyundai's server. Similar attack surfaces exist in other makers with the SiriusXM "smart vehicle" platform [151].

Compared to the previous attacks, our method does not require complicated operations such as reverse engineering of the vehicles' communication protocols and hardware. It also does not require access to the car's OBD ports. Instead, it exploits hidden vulnerabilities in existing smart devices and automation tools, significantly reducing the cost of carrying out safety-critical attacks against vehicles.

## 8. Conclusion

This paper investigated the security vulnerability within automated control chains linking smart glasses and smartphones to Tesla vehicles via voice assistants and apps. We explored the feasibility of an adversary bypassing authentication mechanisms and user interaction to control the automated functionalities maliciously. We demonstrated how attackers can exploit smart glasses as an entry point to manipulate vehicles at the end of the control chain. As automation tools simplify the process for users to connect various devices and services, the potential for adversaries to infiltrate such interconnected and automated systems increases. Our findings highlight the need for manufacturers to become more aware of these vulnerabilities in connected device chains. It is crucial to reevaluate and potentially redesign the security mechanism of interaction methods between humans and wearable devices within automated control chains.

## Acknowledgments

This work was supported in part by the U.S. National Science Foundation under grants OIA-1946231, CNS-2117785, OIA-2229752, CNS-2330264, and CNS-2231682, and by the U.S. Department of Transportation under Grant 69A3552348327 for the CARMEN+ University Transportation Center.

## References

[1] "Intro to Shortcuts on iPhone and iPad," https://support.apple.com/guide/shortcuts/intro-to-shortcuts-apdf22b0444c/ios, 2023, Accessed: 2023-09-05.

[2] "Tool automation. IFTTT for developers," https://ifttt.com/developers, 2023, Accessed: 2023-09-05.

[3] "Intro to home automation in Shortcuts on iPhone or iPad," https://support.apple.com/guide/shortcuts/intro-to-home-automation-apddb94c7489/ios, Accessed: 2023-10-4.

[4] "How to Automate Your Life With Apple's Shortcuts App," https://www.pcmag.com/how-to/automate-your-life-with-apples-shortcuts-app, Accessed: 2023-10-3.

[5] S. A. Kumer, P. Kanakaraja, A. P. Teja, T. H. Sree, and T. Tejaswni, "Smart home automation using IFTTT and google assistant," *Materials Today: Proceedings*, vol. 46, pp. 4070–4076, 2021.

[6] "Tesla now lets you control your car with Apple Shortcuts," https://mashable.com/article/tesla-ios-shortcuts-siri-apple, Accessed: 2023-08-19.

[7] "Tesla's app now supports automation with Apple Shortcuts," https://www.theverge.com/2023/8/19/23838346/tesla-app-apple-ios-shortcuts-automation-update, Accessed: 2023-08-20.

[8] "How to Automate Your Favorite Tesla Features Using Apple's Shortcuts App," https://www.pcmag.com/how-to/how-to-automate-your-favorite-tesla-features-using-apples-shortcuts-app, Accessed: 2023-08-25.

[9] "Tesla releases official API documentation to support third-party apps," https://electrek.co/2023/10/12/tesla-releases-official-api-documentation-support-third-party-apps/.

[10] "Tesla developer documentation," https://www.tesla.com/developer-docs, Accessed: 2023-10-30.

[11] "Global Smart Glasses Market Report 2023," https://www.prnewswire.com/news-releases/global-smart-glasses-market-report-2023-featuring-luxottica-bose-fastrack-huawei-technologies-oakley-razer--musiclens-301738309.html, Accessed: 2023-9-1.

[12] "Global Smart Glasses Market Report 2023: High Demand from United States, China, Japan, Germany - Trends, Opportunities & Forecasts to 2028," https://finance.yahoo.com/news/global-smart-glasses-market-report-104800618.html, Accessed: 2023-8-3.

[13] N. M. Kumar, N. K. Singh, and V. Peddiny, "Wearable smart glass: Features, applications, current progress and challenges," in *2018 second international conference on green computing and internet of things (ICGCIoT)*. IEEE, 2018, pp. 577–582.

[14] L.-H. Lee and P. Hui, "Interaction methods for smart glasses: A survey," *IEEE access*, vol. 6, pp. 28 712–28 732, 2018.

[15] T. Chen and M.-C. Chiu, "Smart technologies for assisting the life quality of persons in a mobile environment: a review," *Journal of Ambient Intelligence and Humanized Computing*, vol. 9, pp. 319–327, 2018.

[16] "Use Google Home or Assistant to control your Tesla," https://help.tessie.com/article/29-use-google-home-assistant-tesla, Accessed: 2023-09-19.

[17] G. Zhang, C. Yan, X. Ji, T. Zhang, T. Zhang, and W. Xu, "Dolphinattack: Inaudible voice commands," in *Proceedings of the 2017 ACM SIGSAC conference on computer and communications security*, 2017, pp. 103–117.

[18] R. Iijima, S. Minami, Z. Yunao, T. Takehisa, T. Takahashi, Y. Oikawa, and T. Mori, "Audio hotspot attack: An attack on voice assistance systems using directional sound beams," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, 2018, pp. 2222–2224.

[19] T. Sugawara, B. Cyr, S. Rampazzi, D. Genkin, and K. Fu, "Light commands: Laser-Based audio injection attacks on Voice-Controllable systems," in *29th USENIX Security Symposium (USENIX Security 20)*, 2020, pp. 2631–2648.

[20] Q. Yan, K. Liu, Q. Zhou, H. Guo, and N. Zhang, "Surfingattack: Interactive hidden attack on voice assistants using ultrasonic guided waves," in *Network and Distributed Systems Security (NDSS) Symposium*, 2020.

[21] S. Ahmed, I. Shumailov, N. Papernot, and K. Fawaz, "Towards more robust keyword spotting for voice assistants," in *31st USENIX Security Symposium (USENIX Security 22)*, 2022, pp. 2655–2672.

[22] R. He, X. Ji, X. Li, Y. Cheng, and W. Xu, ""OK, Siri" or "Hey, Google": Evaluating Voiceprint Distinctiveness Via Content-based PROLE Score," in *Proceedings of the 31th USENIX Security Symposium*, 2022.

[23] E. Marasco and A. Ross, "A survey on antispoofing schemes for fingerprint recognition systems," *ACM Computing Surveys (CSUR)*, vol. 47, no. 2, pp. 1–36, 2014.

[24] "Protecting Against Fingerprint Spoofing in Mobile Devices ," https://www.synaptics.com/sites/default/files/sentrypoint-anti-spoofing-wp.pdf, Accessed: 2023-10-13.

[25] "Smart Home — IFTTT ," https://ifttt.com/solutions/smart-home, Accessed: 2023-10-8.

[26] N. Basoglu, A. E. Ok, and T. U. Daim, "What will it take to adopt smart glasses: A consumer choice based review?" *Technology in Society*, vol. 50, pp. 50–56, 2017.

[27] "Vue: Your everyday smart glasses," https://vueglasses.com/, Accessed: 2023-8-24.

[28] "Bluetooth Sunglasses - Razer Anzu Smart Glasses," https://www.razer.com/mobile-accessories/razer-anzu-lenses, Accessed: 2023-9-4.

[29] D. F. Kune, J. Backes, S. S. Clark, D. Kramer, M. Reynolds, K. Fu, Y. Kim, and W. Xu, "Ghost talk: Mitigating emi signal injection attacks against analog sensors," in *2013 IEEE Symposium on Security and Privacy*. IEEE, 2013, pp. 145–159.

[30] K. S. Abhishek, L. C. F. Qubeley, and D. Ho, "Glove-based hand gesture recognition sign language translator using capacitive touch sensor," in *2016 IEEE international conference on electron devices and solid-state circuits (EDSSC)*. IEEE, 2016, pp. 334–337.

[31] M. Schmitz, M. Khalilbeigi, M. Balwierz, R. Lissermann, M. Mühlhäuser, and J. Steimle, "Capricate: A fabrication pipeline to design and 3D print capacitive touch sensors for interactive objects," in *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*, 2015, pp. 253–258.

[32] Y. L. E. I. Daewon, "Smart glass," Feb. 20 2018, uS Patent 9,897,808.

[33] D. J. Matthies, C. Weerasinghe, B. Urban, and S. Nanayakkara, "Capglasses: Untethered capacitive sensing with smart glasses," in *Proceedings of the Augmented Humans International Conference 2021*, 2021, pp. 121–130.

[34] "How capacitive touch sensing revolutionized the mobile device market," https://nelson-miller.com/how-capacitive-touch-sensing-revolutionized-the-mobile-device-market/, Accessed: 2024-3-02.

[35] "Touch sensors: A comprehensive guide," https://www.linkedin.com/pulse/touch-sensors-comprehensive-guide-electrical-hub-v3yqf/, Accessed: 2023-8-12.

[36] "Capacitive sensing basics," https://software-dl.ti.com/msp430/msp430_public_sw/mcu/msp430/CapTIvate_Design_Center/1_83_00_08/exports/docs/users_guide/html/CapTIvate_Technology_Guide_html/markdown/ch_basics.html#, Accessed: 2023-11-12.

[37] "Texas Instrument,CapTIvate™ Technology Guide," https://software-dl.ti.com/msp430/msp430_public_sw/mcu/msp430/CapTIvate_Design_Center/latest/exports/docs/users_guide/html/CapTIvate_Technology_Guide_html/markdown/index.html, Accessed: 2024-3-01.

[38] "Complete guide to capacitive touch sensors,chapter 2 operational principles of capacitive touch sensors," https://https://fieldscale.com/learn-capacitive-sensing/intro-to-capacitive-sensors-electrostatics/download-chapter-2/, Accessed: 2023-11-12.

[39] "Self-capacitance measurement," https://onlinedocs.microchip.com/pr/GUID-A8A0085D-58D1-4E41-A07D-B93BFDE11AFE-en-US-4/index.html?GUID-057D7429-BF1E-4084-A4BA-296ABD29CE00, Accessed: 2023-10-29.

[40] "RAZER ANZU - SMART GLASSES," https://www.razer.com/mobile-wearables/razer-anzu-smart-glasses, Accessed: 2023-09-15.

[41] H. Nakajima and T. Yashiro, "i-Automator: A Framework for Intelligent IoT Automation using Machine Learning," in *2022 IEEE 4th Global Conference on Life Sciences and Technologies (LifeTech)*. IEEE, 2022, pp. 125–127.

[42] "Use Shortcuts to automate tasks on iPhone," https://support.apple.com/guide/iphone/shortcuts-iph47e1c9d7d/ios, Accessed: 2023-09-10.

[43] "What is IFTTT? Build Codeless Workflows Across Cloud Services," https://www.itprotoday.com/no-codelow-code/what-ifttt-build-codeless-workflows-across-cloud-services#close-modal, Accessed: 2023-11-1.

[44] M. Abdou, A. M. Ezz, and I. Farag, "Digital automation platforms comparative study," in *2021 4th International Conference on Information and Computer Technologies (ICICT)*. IEEE, 2021, pp. 279–286.

[45] "What is Tessie?" https://developer.tessie.com/docs, Accessed: 2023-09-18.

[46] "Porsche Developer Hub ... to deliver high-quality Porsche APIs and build applications across VW Group," https://developerhub.porsche.io/, Accessed: 2023-10-11.

[47] "16 Car Makers and Their Vehicles Hacked via Telematics, APIs, Infrastructure," https://www.securityweek.com/16-car-makers-and-their-vehicles-hacked-telematics-apis-infrastructure/, Accessed: 2023-10-21.

[48] X. Ji, J. Zhang, S. Jiang, J. Li, and W. Xu, "Capspeaker: Injecting voices to microphones via capacitors," in *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, 2021, pp. 1915–1929.

[49] S. Maruyama, S. Wakabayashi, and T. Mori, "Tap'n ghost: A compilation of novel attack techniques against smartphone touchscreens," in *2019 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2019, pp. 620–637.

[50] Y. Jiang, X. Ji, K. Wang, C. Yan, R. Mitev, A.-R. Sadeghi, and W. Xu, "Wight: Wired ghost touch attack on capacitive touchscreens," in *2022 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2022, pp. 984–1001.

[51] "Get personal results on headphones," https://support.google.com/assistant/answer/9907979?hl=en, Accessed: 2023-10-4.

[52] H. Shan, B. Zhang, Z. Zhan, D. Sullivan, S. Wang, and Y. Jin, "Invisible finger: Practical electromagnetic interference attack on touchscreen-based electronic devices," in *2022 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2022, pp. 1246–1262.

[53] K. Wang, R. Mitev, C. Yan, X. Ji, A.-R. Sadeghi, and W. Xu, "GhostTouch: Targeted attacks on touchscreens without physical touch," in *31st USENIX Security Symposium (USENIX Security 22)*, 2022, pp. 1543–1559.

[54] J. Selvaraj, G. Y. Dayanıklı, N. P. Gaunkar, D. Ware, R. M. Gerdes, and M. Mina, "Electromagnetic induction attacks against embedded systems," in *Proceedings of the 2018 on Asia Conference on Computer and Communications Security*, 2018, pp. 499–510.

[55] Y. Tu, S. Rampazzi, B. Hao, A. Rodriguez, K. Fu, and X. Hei, "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*, 2019, pp. 2301–2315.

[56] G. Y. Dayanikli, R. R. Hatch, R. M. Gerdes, H. Wang, and R. Zane, "Electromagnetic sensor and actuator attacks on power converters for electric vehicles," in *2020 IEEE Security and Privacy Workshops (SPW)*. IEEE, 2020, pp. 98–103.

[57] J.-M. Redouté and M. Steyaert, *EMC of analog integrated circuits*. Springer Science & Business Media, 2009.

[58] C. Yan, H. Shin, C. Bolton, W. Xu, Y. Kim, and K. Fu, "Sok: A minimalist approach to formalizing analog sensor security," in *2020 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2020, pp. 233–248.

[59] I. Giechaskiel and K. Rasmussen, "Taxonomy and challenges of out-of-band signal injection attacks and defenses," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 1, pp. 645–670, 2019.

[60] Y. Tu, V. S. Tida, Z. Pan, and X. Hei, "Transduction shield: A low-complexity method to detect and correct the effects of EMI injection attacks on sensors," in *Proceedings of the 2021 ACM Asia Conference on Computer and Communications Security*, 2021, pp. 901–915.

[61] "RFSpace LPDAMAX Wide-band PCB Log Periodic Antenna ," http://rfspace.com/RFSPACE/Antennas_files/LPDA-MAX.pdf, 2018, Accessed: 2023-9-29.

[62] "Minicircuits ZHL-20W-13+ amplifier." https://www.minicircuits.com/pdfs/ZHL-20W-13+.pdf, 2021, Accessed: 2023-9-29.

[63] "Capacitive touch technology," https://www.bareconductive.com/blogs/resources/make-a-basic-capacitive-sensor-with-electric-paint-and-arduino, Accessed: 2023-09-22.

[64] "Create a Capacitive Sensor with Arduino," https://www.aranacorp.com/en/create-a-capacitive-sensor-with-arduino/, Accessed: 2023-09-22.

[65] "Capacitive touch technology," https://www.geeksforgeeks.org/capacitive-touch-technology/, Accessed: 2023-09-22.

[66] “Circuits and techniques for implementing capacitive touch sensing,” https://www.allaboutcircuits.com/technical-articles/circuits-and-techniques-for-implementing-capacitive-touch-sensing/, Accessed: 2023-09-22.

[67] “Touch sensor application note,” https://github.com/ESP32DE/esp-iot-solution-1/blob/master/documents/touch_pad_solution/touch_sensor_design_en.md, Accessed: 2023-10-29.

[68] “Capacitive sensor microcontrollers application note,” https://www.renesas.com/us/en/document/apn/capacitive-sensor-microcontrollers-ctsu-capacitive-touch-introduction-guide, Accessed: 2023-10-29.

[69] “1 key touch pad detector ic – ttp223,” https://files.seeedstudio.com/wiki/Grove-Touch_Sensor/res/TTP223.pdf, Accessed: 2023-10-29.

[70] “Colpitts oscillator,” https://en.wikipedia.org/wiki/Colpitts_oscillator, Accessed: 2023-9-5.

[71] “High-voltage generator coil,” https://www.aliexpress.com/i/3256802783304995.html, Accessed: 2023-12-06.

[72] “Tesla coil. High power generator.” https://www.aliexpress.us/item/2255800237813933.html, Accessed: 2023-12-06.

[73] “Introduction to Siri Shortcuts,” https://developer.apple.com/videos/play/wwdc2018/211/, Accessed: 2023-10-4.

[74] “Apple Developers: App intents,” https://developer.apple.com/documentation/appintents, Accessed: 2023-10-20.

[75] “Apple Developers: App Shortcuts,” https://developer.apple.com/documentation/appintents/app-shortcuts, Accessed: 2023-10-4.

[76] “Tessie API references,” https://developer.tessie.com/reference/, Accessed: 2023-09-18.

[77] “Naturalreader,” https://www.naturalreaders.com/, Accessed: 2023-7-1.

[78] “Traffic noises (FreeSound),” https://freesound.org/people/kyles/sounds/407089/, Accessed: 2023-10-20.

[79] “Conversation noises (FreeSound),” https://freesound.org/people/rampartian/sounds/236786/, Accessed: 2023-10-20.

[80] T. P. McAlexander, R. R. Gershon, and R. L. Neitzel, “Street-level noise in an urban setting: assessment and contribution to personal exposure,” *Environmental Health*, vol. 14, no. 1, pp. 1–10, 2015.

[81] “Multifunctional EMP Generator - High Frequency Electromagnetic Pulse Device with Fingerprint Lock and Lightbulb Detection Capabilities,” https://www.walmart.com/ip/Multifunctional-EMP-Generator-High-Frequency-Electromagnetic-Pulse-Device-with-Fingerprint-Lock-and-Lightbulb-Detection-Capabilities/5318217223, Accessed: 2023-12-06.

[82] “EWA A109mini Bluetooth Speaker with bass Radiator,” https://www.amazon.com/A109mini-Bluetooth-Radiator-Enhanced-Impactive/dp/B07B62BF2R/, Accessed: 2023-12-01.

[83] “Can a Tesla Be Stolen? Straight Answers,” https://www.carparts.com/blog/can-a-tesla-be-stolen-straight-answers/, Accessed: 2023-9-19.

[84] H. Feng, K. Fawaz, and K. G. Shin, “Continuous authentication for voice assistants,” in *Proceedings of the 23rd Annual International Conference on Mobile Computing and Networking*, 2017, pp. 343–355.

[85] H. Farrukh, M. O. Ozmen, F. K. Ors, and Z. B. Celik, “One key to rule them all: Secure group pairing for heterogeneous iot devices,” in *2023 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2023, pp. 3026–3042.

[86] M. Zhou, Z. Qin, X. Lin, S. Hu, Q. Wang, and K. Ren, “Hidden voice commands: Attacks and defenses on the VCS of autonomous driving cars,” *IEEE Wireless Communications*, vol. 26, no. 5, pp. 128–133, 2019.

[87] J. Liu, W. Song, L. Shen, J. Han, and K. Ren, “Secure user verification and continuous authentication via earphone imu,” *IEEE Transactions on Mobile Computing*, 2022.

[88] Y. Wang, W. Cai, T. Gu, W. Shao, Y. Li, and Y. Yu, “Secure your voice: An oral airflow-based continuous liveness detection for voice assistants,” *Proceedings of the ACM on interactive, mobile, wearable and ubiquitous technologies*, vol. 3, no. 4, pp. 1–28, 2019.

[89] C. Yan, X. Ji, K. Wang, Q. Jiang, Z. Jin, and W. Xu, “A survey on voice assistant security: Attacks and countermeasures,” *ACM Computing Surveys*, vol. 55, no. 4, pp. 1–36, 2022.

[90] C. Yan, Y. Long, X. Ji, and W. Xu, “The catcher in the field: A fieldprint based spoofing detection for text-independent speaker verification,” in *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, 2019, pp. 1215–1229.

[91] L. Zhang, S. Tan, and J. Yang, “Hearing your voice is not enough: An articulatory gesture based liveness detection for voice authentication,” in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, 2017, pp. 57–71.

[92] “FRAME STEMPO,” https://assets.bose.com/content/dam/Bose_DAM/Web/consumer_electronics/global/products/wearables/bose_frames_tempo/pdf/856768_og_frames-tempo_en.pdf, Accessed: 2023-8-2.

[93] T. Arif, A. Javed, M. Alhameed, F. Jeribi, and A. Tahir, “Voice spoofing countermeasure for logical access attacks detection,” *IEEE Access*, vol. 9, pp. 162 857–162 868, 2021.

[94] J. Zhou, T. Hai, D. N. Jawawi, D. Wang, E. Ibeke, and C. Biamba, “Voice spoofing countermeasure for voice replay attacks using deep learning,” *Journal of Cloud Computing*, vol. 11, no. 1, p. 51, 2022.

[95] Y. Zhang and K. Rasmussen, “Detection of electromagnetic interference attacks on sensor systems,” in *2020 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2020, pp. 203–216.

[96] M. M. Ahmadpanah, D. Hedin, M. Balliu, L. E. Olsson, and A. Sabelfeld, “{SandTrap}: Securing {JavaScript-driven}{Trigger-Action} platforms,” in *30th USENIX Security Symposium (USENIX Security 21)*, 2021, pp. 2899–2916.

[97] Q. Jiang, Y. Ren, Y. Long, C. Yan, Y. Sun, X. Ji, K. Fu, and W. Xu, “Ghosttype: The limits of using contactless electromagnetic interference to inject phantom keys into analog circuits of keyboards,” in *Network and Distributed Systems Security (NDSS) Symposium*, 2024.

[98] J.-H. Jang, M. Cho, J. Kim, D. Kim, and Y. Kim, “Paralyzing drones via emi signal injection on sensory communication channels.” in *NDSS*, 2023.

[99] S.-H. Min, H. Jung, O. Kwon, M. Sattorov, S. Kim, S.-H. Park, D. Hong, S. Kim, C. Park, B. H. Hong *et al.*, “Analysis of electromagnetic pulse effects under high-power microwave sources,” *IEEE Access*, vol. 9, pp. 136 775–136 791, 2021.

[100] G. Ni, B. Gao, and J. Lu, “Research on high power microwave weapons,” in *2005 Asia-Pacific Microwave Conference Proceedings*, vol. 2. IEEE, 2005, pp. 4–pp.

[101] Q. Xia, Q. Chen, and S. Xu, “Near-Ultrasound Inaudible Trojan (NUIT): Exploiting Your Speaker to Attack Your Microphone, usenix security 2023,” 2023.

[102] “Complete guide to the Shortcuts app on macOS ,” https://www.xda-developers.com/guide-shortcuts-macos/, Accessed: 2023-10-4.

[103] “Drone Mobile API,” https://github.com/Hacksore/drone-mobile, Accessed: 2023-9-17.

[104] “Compustar: Remote Start and Vehicle Security Systems,” https://www.compustar.com/, Accessed: 2023-9-22.

[105] “World health organization. Radiation: Electromagnetic fields.” https://www.who.int/news-room/questions-and-answers/item/radiation-electromagnetic-fields, Accessed: 2023-1-01.

[106] X. Yuan, Y. Chen, A. Wang, K. Chen, S. Zhang, H. Huang, and I. M. Molloy, “All your alexa are belong to us: A remote voice control attack against echo,” in *2018 IEEE global communications conference (GLOBECOM)*. IEEE, 2018, pp. 1–6.

[107] T. Vaidya, Y. Zhang, M. Sherr, and C. Shields, "Cocaine noodles: exploiting the gap between human and machine speech recognition," in *9th USENIX Workshop on Offensive Technologies (WOOT 15)*, 2015.

[108] N. Carlini, P. Mishra, T. Vaidya, Y. Zhang, M. Sherr, C. Shields, D. Wagner, and W. Zhou, "Hidden voice commands," in *25th USENIX security symposium (USENIX security 16)*, 2016, pp. 513–530.

[109] X. Yuan, Y. Chen, Y. Zhao, Y. Long, X. Liu, K. Chen, S. Zhang, H. Huang, X. Wang, and C. A. Gunter, "CommanderSong: A systematic approach for practical adversarial voice recognition," in *27th USENIX security symposium (USENIX security 18)*, 2018, pp. 49–64.

[110] H. Abdullah, W. Garcia, C. Peeters, P. Traynor, K. R. Butler, and J. Wilson, "Practical hidden voice attacks against speech and speaker recognition systems," *arXiv preprint arXiv:1904.05734*, 2019.

[111] N. Carlini and D. Wagner, "Audio adversarial examples: Targeted attacks on speech-to-text," in *2018 IEEE security and privacy workshops (SPW)*. IEEE, 2018, pp. 1–7.

[112] L. Schönherr, K. Kohls, S. Zeiler, T. Holz, and D. Kolossa, "Adversarial attacks against automatic speech recognition systems via psychoacoustic hiding," *arXiv preprint arXiv:1808.05665*, 2018.

[113] T. Chen, L. Shangguan, Z. Li, and K. Jamieson, "Metamorph: Injecting inaudible commands into over-the-air voice controlled systems," in *Network and Distributed Systems Security (NDSS) Symposium*, 2020.

[114] J. Esteves and C. Kasmi, "You don't hear me but your phone's voice interface does," 06 2015.

[115] C. Kasmi and J. L. Esteves, "Whisper in the wire: Voice command injection reloaded," *Hack In Paris*, 2016.

[116] D. Dai, Z. An, and L. Yang, "Inducing wireless chargers to voice out for inaudible command attacks," in *2023 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2023, pp. 1789–1806.

[117] N. Roy, H. Hassanieh, and R. Roy Choudhury, "Backdoor: Making microphones hear inaudible sounds," in *Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services*, 2017, pp. 2–14.

[118] N. Roy, S. Shen, H. Hassanieh, and R. R. Choudhury, "Inaudible voice commands: The Long-Range attack and defense," in *15th USENIX Symposium on Networked Systems Design and Implementation (NSDI 18)*, 2018, pp. 547–560.

[119] R. Iijima, S. Minami, Y. Zhou, T. Takehisa, T. Takahashi, Y. Oikawa, and T. Mori, "Audio hotspot attack: An attack on voice assistance systems using directional sound beams and its feasibility," *IEEE Transactions on Emerging Topics in Computing*, vol. 9, no. 4, pp. 2004–2018, 2019.

[120] B. Cyr, T. Sugawara, and K. Fu, "Why lasers inject perceived sound into mems microphones: Indications and contraindications of photoacoustic and photoelectric effects," in *2021 IEEE Sensors*. IEEE, 2021, pp. 1–4.

[121] G. Li, Z. Cao, and T. Li, "Echoattack: Practical inaudible attacks to smart earbuds," in *Proceedings of the 21st Annual International Conference on Mobile Systems, Applications and Services*, 2023, pp. 383–396.

[122] W. Diao, X. Liu, Z. Zhou, and K. Zhang, "Your voice assistant is mine: How to abuse speakers to steal information and control your phone," in *Proceedings of the 4th ACM Workshop on Security and Privacy in Smartphones & Mobile Devices*, 2014, pp. 63–74.

[123] P. J. Young, J. H. Jin, S. Woo, and D. H. Lee, "Badvoice: Soundless voice-control replay attack on modern smartphones," in *2016 Eighth International Conference on Ubiquitous and Future Networks (ICUFN)*. IEEE, 2016, pp. 882–887.

[124] Y. Wang, H. Guo, and Q. Yan, "Ghosttalk: Interactive attack on smartphone voice system through power line," *arXiv preprint arXiv:2202.02585*, 2022.

[125] M. I. Hossen, Y. Tu, and X. Hei, "A first look at the security of eeg-based systems and intelligent algorithms under physical signal injections," in *Proceedings of the 2023 Secure and Trustworthy Deep Learning Systems Workshop*, 2023, pp. 1–8.

[126] Y. Long, S. Rampazzi, T. Sugawara, and K. Fu, "Protecting covid-19 vaccine transportation and storage from analog cybersecurity threats," *Biomedical Instrumentation & Technology*, vol. 55, no. 3, pp. 112–117, 2021.

[127] K. Fu and W. Xu, "Risks of trusting the physics of sensors," *Communications of the ACM*, vol. 61, no. 2, pp. 20–23, 2018.

[128] H. Zhu, Z. Yu, W. Cao, N. Zhang, and X. Zhang, "Powertouch: A security objective-guided automation framework for generating wired ghost touch attacks on touchscreens," in *Proceedings of the 41st IEEE/ACM International Conference on Computer-Aided Design*, 2022, pp. 1–9.

[129] Y. Jiang, X. Ji, K. Wang, C. Yan, R. Mitev, A.-R. Sadeghi, and W. Xu, "Marionette: Manipulate your touchscreen via a charging cable," *IEEE Transactions on Dependable and Secure Computing*, 2023.

[130] "Technical Advisory – Tesla BLE Phone-as-a-Key Passive Entry Vulnerable to Relay Attacks," https://research.nccgroup.com/2022/05/15/technical-advisory-tesla-ble-phone-as-a-key-passive-entry-vulnerable-to-relay-attacks/, Accessed: 2023-09-15.

[131] "Rolling pwn attack," https://rollingpwn.github.io/rolling-pwn/, Accessed: 2023-09-15.

[132] J. Wang, K. Lounis, and M. Zulkernine, "Cskes: a context-based secure keyless entry system," in *2019 IEEE 43rd Annual Computer Software and Applications Conference (COMPSAC)*, vol. 1. IEEE, 2019, pp. 817–822.

[133] S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham, S. Savage, K. Koscher, A. Czeskis, F. Roesner, and T. Kohno, "Comprehensive experimental analyses of automotive attack surfaces," in *20th USENIX security symposium (USENIX Security 11)*, 2011.

[134] F. D. Garcia, D. Oswald, T. Kasper, and P. Pavlidès, "Lock it and still lose it—on the (In) Security of automotive remote keyless entry systems," in *25th USENIX security symposium (USENIX Security 16)*, 2016.

[135] S. Kamkar, "Drive it like you hacked it: New attacks and tools to wirelessly steal cars," *Presentation at DEFCON*, vol. 23, p. 10, 2015.

[136] S. G. Philipsen, B. Andersen, and B. Singh, "Threats and attacks to modern vehicles," in *2021 IEEE International Conference on Internet of Things and Intelligence Systems (IoTaIS)*. IEEE, 2021, pp. 22–27.

[137] "R7-2017-02: Hyundai Blue Link Potential Info Disclosure," https://www.rapid7.com/blog/post/2017/04/25/r7-2017-02-hyundai-blue-link-potential-info-disclosure-fixed/, Accessed: 2023-09-01.

[138] C. Anliker, G. Camurati, and S. Čapkun, "Time for Change: How Clocks Break UWB Secure Ranging," in *32nd USENIX Security Symposium (USENIX Security 23)*, 2023, pp. 19–36.

[139] X. Xie, K. Jiang, R. Dai, J. Lu, L. Wang, Q. Li, and J. Yu, "Access Your Tesla without Your Awareness: Compromising Keyless Entry System of Model 3." in *NDSS*, 2023.

[140] P. Leu, G. Camurati, A. Heinrich, M. Roeschlin, C. Anliker, M. Hollick, S. Capkun, and J. Classen, "Ghost Peak: Practical Distance Reduction Attacks Against {HRP}{UWB} Ranging," in *31st USENIX Security Symposium (USENIX Security 22)*, 2022, pp. 1343–1359.

[141] "New Tesla Key Card Vulnerability Lets Hackers Silently Steal Your Ride," https://www.howtogeek.com/120570/new-tesla-key-card-vulnerability-lets-hackers-silently-steal-your-ride/, Accessed: 2023-9-25.

[142] S. Nie, L. Liu, and Y. Du, "Free-fall: Hacking Tesla from wireless to can bus," *Briefing, Black Hat USA*, vol. 25, no. 1, p. 16, 2017.

[143] S. Nie, L. Liu, Y. Du, and W. Zhang, “Over-the-air: How we remotely compromised the gateway, BCM, and autopilot ECUs of Tesla cars,” *Briefing, Black Hat USA*, vol. 91, 2018.

[144] S. Mazloom, M. Rezaeirad, A. Hunter, and D. McCoy, “A Security Analysis of an In-Vehicle Infotainment and App Platform,” in *10th USENIX Workshop on Offensive Technologies (WOOT 16)*, 2016.

[145] H. J. Jo, W. Choi, S. Y. Na, S. Woo, and D. H. Lee, “Vulnerabilities of android OS-based telematics system,” *Wireless Personal Communications*, vol. 92, pp. 1511–1530, 2017.

[146] D. Klinedinst and C. King, “On board diagnostics: Risks and vulnerabilities of the connected vehicle,” *CERT Coordination Center, Tech. Rep*, 2016.

[147] H. Wen, Q. Zhao, Q. A. Chen, and Z. Lin, “Automated cross-platform reverse engineering of can bus commands from mobile apps,” in *Proceedings of the 27th Annual Network and Distributed System Security Symposium (NDSS’20)*, San Diego, CA, February 2020.

[148] H. Wen, Q. A. Chen, and Z. Lin, “Plug-N-Pwned: Comprehensive vulnerability analysis of OBD-II dongles as a new Over-the-Air attack surface in automotive IoT,” in *29th USENIX Security Symposium (USENIX Security 20)*, 2020, pp. 949–965.

[149] “blackhat Europe 2022: Back-connect to the Connected Car. Search for Vulnerabilities in the VW Electric Car.” https://www.blackhat.com/eu-22/briefings/schedule/#back-connect-to-the-connected-car-search-for-vulnerabilities-in-the-vw-electric-car-29506, Accessed: 2024-3-01.

[150] “Tesla Model 3 Hacked in Less Than 2 Minutes at Pwn2Own Contest,” https://www.darkreading.com/vulnerabilities-threats/tesla-model-3-hacked-2-minutes-pwn2own-contest.

[151] “Hyundai app bugs allowed hackers to remotely unlock, start cars,” https://www.bleepingcomputer.com/news/security/hyundai-app-bugs-allowed-hackers-to-remotely-unlock-start-cars/, Accessed: 2023-09-15.

## Appendix A. Teardown and Experiment Setting Photos

Fig. 13 shows the teardown photos of two pairs of smart glasses. Fig. 14 shows the prototype touch sensing circuit and EMI attack setting. The Arduino’s digital Pin D4 is HIGH when charging the circuit. We use the analog Pin A0 to measure the detected voltage. In each charging cycle, when the detected voltage exceeds a threshold, the circuit records the charge time, and the digital Pin D4 turns to LOW for 10 ms to discharge the circuit.

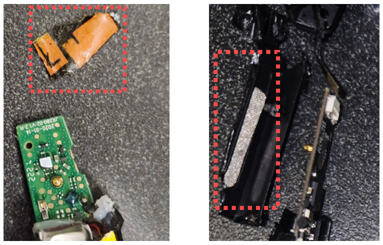


Figure 13: Teardown photos of Razer Anzu smart glasses (left) and Eyewear pro smart glasses (right) show the Capacitive touch sensing pads in smart glasses.

## Appendix B. Customized Shortcuts

In the customized control chain, we use an automation tool (Apple Shortcuts) to trigger another automation tool (IFTTT) to access the server APIs. Specifically, we configure the IFTTT applets to be invoked by Apple Shortcuts and customize the Shortcuts to trigger the specific IFTTT applets. The IFTTT applets then send Webhook requests to the Tessie API, which communicates the Tesla API to control the car functionalities. Table 7 shows the customized Shortcuts commands we have set up. We test them in attacks exploiting smart glasses to invoke Shortcuts and subsequently trigger IFTTT Applets on screen-locked iPhones to execute the customized control chain.

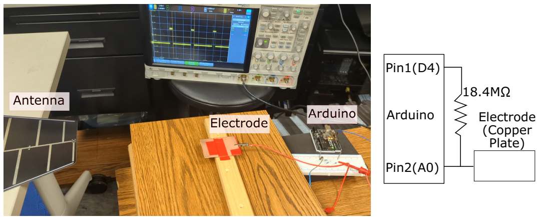


Figure 14: Left: Experiment settings of EMI attacks on the prototype touch sensing circuit. Right: Circuit diagram of the prototype touch sensing circuit. D4 is the output pin. A0 is the input pin.

TABLE 7: Tested customized shortcuts commands.

| Apple Shortcuts | |
|---|---|
| C1 | Unlock My Tesla |
| C2 | Lock My Tesla |
| C3 | Open Front Trunk |
| C4 | Open Rear Trunk |
| C5 | Vent Windows |
| C6 | Close Windows |
| C7 | Remote Start |
| C8 | Open Charge Port |
| C9 | Close Charge Port |

## Appendix C. Tested Outdoor Attack Scenarios

We validate our attacks on four outdoor scenarios. Among them, the coffee shop parking lot is adjacent to a city street with the background noise reaching to almost 70 dBA during our testing period. The Tesla supercharger station is near a highway. Its background noise reaches 66 dBA in off-peak period. The remaining two are relatively further from main streets and the noise levels are lower.

Figure 15: Photos of four outdoor scenarios we have tested.

## Appendix D. Meta-Review

The following meta-review was prepared by the program committee for the 2024 IEEE Symposium on Security and Privacy (S&P) as part of the review process as detailed in the call for papers.

### D.1. Summary

The paper develops an attack methodology that exploits a control chain in a cyber-physical system. The attack manipulates smart glasses at the start of the chain, where the attacker generates Electromagnetic Interference (EMI) signals directed at smart glasses to falsely trigger Voice Assistant (VA). Progressing further into the control chain, the attack leverages automated control systems such as Apple's Shortcuts with Tesla app or IFTTT with Tessie to perform malicious actions like unlocking a Tesla at the end of the chain.

### D.2. Scientific Contributions

- Identifies an Impactful Vulnerability
- Provides a Valuable Step Forward in an Established Field

### D.3. Reasons for Acceptance

1) The paper provides an interesting lesson on how a series of seemingly minor vulnerabilities can be chained together for tangible attacks, unlocking a Tesla by a cheap device emitting EMI.
2) The proposed threat model provides valuable lessons on the importance of authentication and trust policy within and between mobile/wearable devices.
3) The use of EMI to spoof capacitive sensors of smart glasses to activate VA systems emphasizes the need for further research on the security of wearable devices.

### D.4. Noteworthy Concerns

Some reviewers have pointed out that while the chained attack results in a clever end-to-end attack, some of the individual components of the attack are not necessarily novel. For example, EMI attacks on capacitive touch screens are not a novel finding.