Analog Signal Security: Security Threats Caused by Physical Phenomena and Their Countermeasures

アナログ信号セキュリティ: 物理現象によって生じるセキュリティ脅威とその対策

July, 2022

Ryo IIJIMA

飯島　涼

Analog Signal Security: Security Threats Caused by Physical Phenomena and Their Countermeasures

アナログ信号セキュリティ: 物理現象によって生じるセキュリティ脅威とその対策

July, 2022

Waseda University

Graduate School of Fundamental Science and Engineering

Department of Computer Science and Communications Engineering

Research on Networked Systems

Ryo IIJIMA

飯島　涼

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

The downsizing of computers and the decreasing price of sensors have led to the widespread use of IoT devices equipped with sensors. IoT devices reached a market size of $33.06 billion in 2020 and are expected to reach $1.5 trillion by 2030, while the total number of Internet-connected things is expected to reach 24.1 billion [1]. Representative examples of IoT devices include voice assistants that control home appliances by voice and smartwatches that manage a person's health status. With the spread of smartphones and wearable devices connected to the cloud, the number of sensor-equipped IoT devices is expected to increase further.

IoT device has three main elements: sensors, controllers, and actuators. Sensors and actuators are shown as the interface between physical space and cyberspace. Sensors are the interface that inputs the analog signal generated in the physical space to the device, and actuators are the interface that outputs the data processed in the device to the physical space. Controllers analyze the information received from the sensor and operate the actuator according to the results. Data transmission, analysis, and decision-making by controllers can be done through networks.

Although the widespread adoption of IoT devices can result in substantial benefits to society, many security threats that exploit the physical sensors inherent in IoT devices [2, 3, 4, 5, 6, 7, 8, 9, 10] have been identified. These threats are caused by

analog signals present in the physical world. A common problem caused by analog signal threats is that incorrect values injected into sensors can cause devices to make incorrect authentication or control decisions. There are two main types of problems caused by incorrect value injection: (1) Spoofing, and (2) Jamming [11]. A spoofing attack is an attack in which a false signal is intentionally presented to a sensor to mislead its decision making. An injection Attack is a method to input values to a sensor in an incorrect way in order to perform a spoofing attack. For example, in the case of sound signals, the replay attack authenticates by emitting a recorded voice of another person from a loudspeaker. Voice synthesis attack [12, 13] creates a voice for authentication by synthesizing a recorded voice when the voice for authentication cannot be obtained. Voice conversion attack [14] creates a model to convert the attacker's voice into the target's voice. Replay attack and voice synthesize attack both use ordinary dynamic speakers, thus the threat level is low because they can be noticed if other people are in the same environment. Inaudible attack, which attacks without the user's knowledge, has been proposed as an attack with a high threat level. The DolphinAttack [8], which uses ultrasonic waves to achieve voice recognition by the user secretly, and Hidden voice commands [9, 15], which process the noises that cause voice recognition. Analog signals point out threats not only to voice, but also to light, biometric signals, and a wide range of other types of analog signals. LightCommands [6] that perform voice recognition by injecting light signals into a microphone sensor, ECG attacks [16] that use biometric signals obtained from the human body to regenerate the subject's biometric signals, SigR attacks [17] that estimate Photoplethysmogram (PPG) waves from facial videos to break through victim's PPG authentication. These sensors are embedded in medical devices [18] and vehicles, such as self-driving cars [19], and misjudgment due to incorrect sensor values may affect human health and human life.

A novel problem with analog signal threats is that they can occur even if the owner or creator of the device does not intend the attack. Ding et al. report that analog signals from IoT devices pose 162 different threats to other sensors in the same physical space [20]. The same threat was pointed out for radio signals. However, since laws for operating radio signals were established, and technical standards

conformity defined the regulations to be followed at the time of shipment, it was unrealistic to expect threats to appear in the real world. On the other hand, analog signals may pose a threat after shipment because technical standards conformity and other regulations do not specify standards for how analog signals should be output. As the number of IoT devices with sensors and actuators increases, similar threats are expected to increase.

Previous security solutions for threats to sensors have been proposed and implemented mainly for digital signals after A/D conversion on the input side [2, 21, 22, 4, 23, 8, 24, 25, 26, 27, 28]. In [24], when speech is input to a microphone, a feature called pop noise occurs when speech is input from the mouth, is used for detection. In [29], in order to detect voice spoofing attacks, a deep learning model is created for digital signals input to sensors to determine whether or not they are attacks. Furthermore, [28] proposes a detection method by using high-frequency regions that are difficult to represent by speakers as features. For sensors in VR devices, an approach to control is also taken on the input side of the sensor [30].

These countermeasures usually focused on sensor input data regardless of sensor type, presenting the following shortcomings: (1) Attack detection accuracy may be reduced considerably by the noise generated in the physical space [31, 4, 13] and (2) It is difficult to accurately detect an attack that exploits circuit nonlinearity of input-processing data [32, 8, 33] (3) Input-based approaches do not directly block the source of the attack. Security measures against threats in the analog signal domain before digitization have not been considered.

1.2 Objective & Research Targets

Under these situations, this thesis aims to solve the security problems caused by analog signals. We focus on the security threats posed by analog signals to address the problems described above. Specifically, experiments and evaluations were conducted to identify countermeasures against threats posed by analog signals. The areas targeted by this research are described following and shown in Figure 1.1.

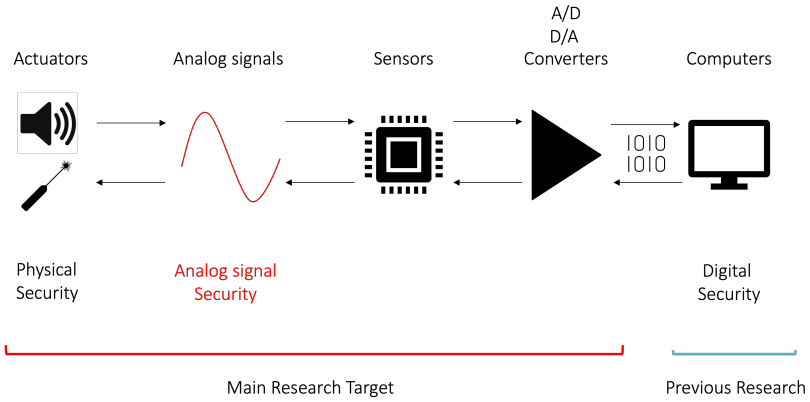


Fig. 1.1 The area of main research targets and previous research.

Analog signal security is defined as the area of security threats and countermeasures caused by analog signals. It is essential to survey previous research on threats posed by sound and light to identify new threats that could be caused. The difference between analog signal security and conventional security techniques is that analog signals do not have explicit attributes. When controlling the output of a digital signal, the object has explicit attributes, making it easy to implement controls such as if-then rules [34, 30]. On the other hand, time series data related to analog signals do not have explicit attributes. In this study, we design a method to explicitly attribute analog signals to facilitate security control in Chapter 3. In this thesis, we focus on the spoofing attack threat using sound signals in analog signal security and discuss its threats and countermeasures in Chapter 2 as audio security.

Sensor security

This research aims to reduce the total number of threats that appear in the real world through a new approach by analyzing the output side of the signal, regardless of the type of analog signal. In Chapter 3, we develop a framework that generalizes standard analysis and countermeasure methods by using analog signals as time-series signals and conduct a case study using audio signals as an example to show that comprehensive countermeasures can be performed regardless of the type of analog signal.

1.3 Contributions

The contributions of this thesis are as follows.

Contribution 1: Evaluating Threat Caused by Analog Signals

We propose a novel attack, called an "Audio Hotspot Attack," which performs an inaudible malicious voice command attack, by targeting voice assistance systems, e.g., smart speakers or in-car navigation systems. The key idea of the approach is to leverage directional sound beams generated from parametric loudspeakers, which emit amplitude-modulated ultrasounds that will be self-demodulated in the air. Our work goes beyond the previous studies of inaudible voice command attack in the following three aspects: (1) the attack can succeed on a long distance (3.5 meters in a small room, and 12 meters in a long hallway), (2) it can control the spot of the audible area by using two directional sound beams, which consist of a carrier wave and a sideband wave, and (3) the proposed attack leverages a physical phenomenon i.e.,non-linearity in the air, to attack voice assistance systems. This study presents and verifies an attack using physical phenomena and reveals a new perspective: the possibility of analog signal threats in physical formulas for the first time. To evaluate the feasibility of the attack, we performed extensive in-lab experiments and a user study involving 20 participants. The results demonstrated that the attack was feasible in a real-world setting. We discussed the extent of the threat, as well as the possible countermeasures against the attack.

Contribution 2: Preventing Threat Caused by Analog Signals

Based on the knowledge obtained from the contribution 1, this work developed a new security framework named Cyber-Physical Firewall (CPFW), which provides a generic and flexible access control mechanism for regulating the malicious analog signals that target cyber-physical system (CPS) devices. This framework enables the defeat of various attacks that make use of malicious analog signals against CPS devices; e.g., stealth voice command injection attack using ultrasonic waves or adversarial examples, or attacks to crash drones in flight using malicious sound waves. Based on relevant previously reported findings, we first defined the requirements

and design specifications of the CPFW framework. In order to detect and regulate analog signals, this study focuses on signal attributes, the acquisition of which has not been explicitly defined in the previous research, and establishes a mechanism for acquiring attributes and a method for detecting and regulating them. Then, we built a prototype CPFW framework and demonstrated its feasibility through extensive performance evaluations and case-study experiments using three real-world attacks; ultrasonic attacks (DolphinAttack), noise attacks (Audio Adversarial Examples), and resonant attacks (WALNUT).

1.4 Outline

The structure of this thesis is as follows. In Chapter 2, we discuss sound signals as a typical threat posed by analog signals and show the potential for new threats based on physical phenomena in the air. In Chapter 3, after identifying the security threats posed by analog signals, such as light, biometric signals, wireless signals, etc. we propose a framework for implementing countermeasures as the Cyber-Physical Firewall. The framework was designed, implemented, and evaluated, and a case study was conducted using sound signals as an example. Chapter 4 describes the limitations in analog signal security and provides an overview of future directions. Chapter 5 summarizes this thesis.

Chapter 2

Contribution1: Evaluating Threat Caused by Analog Signals

2.1 Introduction

Voice assistance systems, such as Siri [35], Google Assistant [36], and Amazon Alexa [37] have become increasingly popular as a means to establish user-friendly human–computer interactions. Voice assistance systems are now supported on various devices, e.g., smartphones/tablets, smart speakers, automobiles, smart homes, smart watches, smart TVs, media boxes, and laptops/desktops. Voice assistance systems can integrate speech recognition to demonstrate various skills such as providing recommendations to restaurants, reading out schedules, and even purchasing products when an appropriate voice command is given.

While these voice assistance systems have clear benefits in daily life activities, they also raise intrinsic security and privacy concerns. One of the most serious security issues related to the use of voice assistance systems is the lack of a rigorous mechanism to guarantee the trustworthiness of the voice source that operates the system. As previous studies have demonstrated [15, 8], voice assistance systems are vulnerable to "inaudible voice command attacks." Here, an attacker can issue voice commands to a voice assistance device unbeknownst to the device owner. For instance, if an attacker generates an inaudible voice command that adjusts the

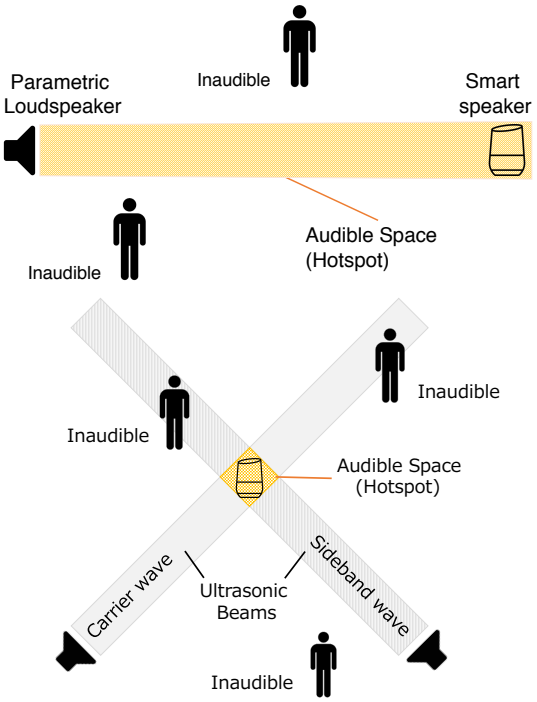


Fig. 2.1 Overview of the Audio Hotspot Attack. Top: Attack with one parametric loudspeaker (linear attack). Bottom: Attack with two parametric loudspeakers (cross attack). In the yellow colored area, you can hear the sound.

volume of the music player set in a car to its maximum, the driver may be surprised or momentarily distracted, thus increasing the likelihood of an accident. Recent studies have leveraged existing vulnerabilities of the device or software. In Ref. [8], the authors found that ultrasound can be used to convey inaudible voice command attacks, by using the vulnerability of the amplifier. Hidden voice commands [15] used the vulnerability of machine learning models that incorrectly recognize noise as normal commands.

We propose a novel inaudible voice attack, named Audio Hotspot Attack, which leverages the *physical phenomena*. In this attack, attackers attempt to input directional sound to voice assistance systems as shown in Figure 2.1. Directional sound is generated by using the nonlinearity of ultrasonic waves in the air. When the modulated ultrasound passes through the air, which acts as a nonlinear medium, the signal

is demodulated into audible sound even if a demodulation circuit is *not* prepared. It is well known that the demodulated sound signals exhibit higher directivity than those emitted from a normal loudspeaker [38, 39]. To generate directional sound, we make use of a parametric loudspeaker, which composes of an array of ultrasound transducers.

The attack proposed in this paper is different from previously proposed attacks in that it leverages physical phenomena that cannot be modified or eliminated. As the previous attacks use vulnerabilities associated with hardware or software, they can be fixed, e.g., by modifying the machine learning algorithm or eliminating the nonlinearity of the microphone. In contrast, the nonlinearity of air is a natural phenomenon, and it is impossible to take measures against it using conventional approaches.

Furthermore, the adoption of parametric loudspeakers enables an attacker to perform a unique form of the attack, called a *cross attack*. As shown at the bottom of Figure 2.1, an attacker sets two parametric loudspeakers in different places and transmits directional sound beams to the target voice assistance device. The two sound beams are inaudible because each sound beam consists of a carrier wave or sideband wave with ultrasound frequency. The sound beams become audible where the two beams cross at a point; i.e., they become an AM sound wave. An attacker can take control of the cross point by adjusting the sources of the two sound beams.

To evaluate the feasibility of the attack, we pose the following research questions:

RQ1: *Is the Audio Hotspot Attack feasible at long distance with off-the-shelf voice assistant devices?*

RQ2: *Does the Audio Hotspot Attack succeed in noisy practical environments?*

RQ3: *Is the attack stealthy for nearby people and unrecognizable for them?*

We aim to answer these questions through extensive experiments and user studies involving 20 participants.

The contributions of this work can be summarized as follows:

- We proposed a novel inaudible voice command attack that targets voice assistance systems, leveraging the directional sound beams to create a “hotspot”

of the attack success area (Section 2.3).

- We carefully designed and controlled our experiments. We used a room and equipment dedicated to acoustic experiments (Section 2.4).
- We demonstrated that the attack could succeed at a long distance. We discovered that the attacks were tolerant of environmental noise. For both devices, the attack success rate remained high at a noise sound pressure level. We showed that the cross attack was also feasible (Section 2.5).
- Through the extensive user studies, we demonstrated that people could not recognize the attacker's voice (Section 2.6).
- We discussed potential threats that may arise in the future as well as the possible countermeasures against the attack (Section 2.7).

To the best of our knowledge, this work is the first to make use of directional sound beams as a means of attacking voice-controlled systems. This perspective sheds new light on security and privacy issues for systems that make use of sound.

2.2 background

In this section, we describe the three key technologies that constitute our attack: the voice assistance system, parametric loudspeakers, and voice presentation attack.

2.2.1 Voice Assistance Systems

Currently, a typical voice assistance system has two action phases for device operation: activation and recognition. In the first phase, a user speaks a specific wake-up word to activate the system, e.g., "OK Google" for Google Assistant, "Alexa" for Amazon Alexa, and "Hey Siri" for Apple Siri. In the second phase, a user transmits a voice command to the system. The system applies speech recognition to the received voice data and executes a command extracted from this data. The available voice commands include common operations such as turning on a light, answering questions, reading the news, or privacy-sensitive operations that access personal resources such as reading out schedules, sending a text message, making a phone

call, or purchasing a product.

Many of the smart speakers today offer speaker recognition functionality so that each person in the household can enjoy the device in a customizable way. For instance, each person using the Amazon Echo can link their own Amazon account to the device. The device identifies each person by leveraging voiceprints to employ biometric verification. To be enrolled in the device's speaker recognition, an owner of the device first needs to register his or her voiceprint, typically by saying a wake-up word multiple times. By comparing the wake-up word against a previously created voiceprint, the voice assistance system verifies a person' s identity. Although a third person who is not registered can still attempt to use the device, his or her usage will be limited to non-personalized common services such as reading news or weather forecasts.

As we will discuss in Section 2.3, speaker recognition technology is vulnerable to voice presentation attacks [4]. These attacks attempt to bypass voice authentication using voice replay/synthesis/conversion technique fraudulently (See Section 2.2.3).

2.2.2 Mechanism of parametric loudspeakers

A parametric loudspeaker can generate directional sound using ultrasound. It consists of an array of many ultrasound transducers installed in parallel [40]. Figure 2.2 presents a parametric loudspeaker used throughout the experiments. Each ultrasonic transducer transmits ultrasound that modulates the original sound wave with amplitude modulation (AM). The generated ultrasound is self-demodulated in the air and becomes audible even if we do not prepare a demodulation circuit (called self-demodulation [38]). Next, we present the self-demodulation mechanism, also known as the *parametric phenomenon*.

Let $p = p(x, t)$ be the sound pressure caused by sound wave originating from a parametric loudspeaker, where x is the distance from the loudspeaker and t is time. As the sound wave is AM-modulated, it has three major frequencies, i.e., carrier frequency, f_c, and adjacent sideband, f_{s_-}, f_{s_+} where $f_{s_-} = f_c - f_m$, $f_{s_+} = f_c + f_m$. f_m represents the frequency of the sound wave to be injected by an attacker. We

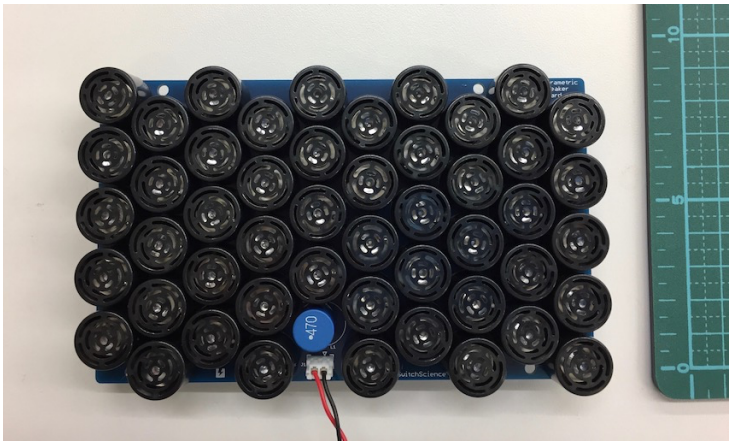


Fig. 2.2 A parametric loudspeaker. This loudspeaker can generate directional sound. It consists of an array of ultrasonic-emitting loudspeakers arranged in a grid. A parametric loudspeaker emits sounds on a narrow spatial range containing a targeted device.

focus on lower sideband to simplify. Primary wave p is expressed as

$$p = p_c \sin(2\pi f_c t') + p_{s\_} \sin\left(2\pi f_{s\_} t'\right) \tag{2.1}$$

p_c and $p_{s\_}$ are the amplitudes of the carrier wave and the sideband wave, respectively. where $t' = t - x/c_0$ is a *retarded time*; the retarded time is the time when the sound wave began to propagate from the sound source.

Burger's equation is one of the fluid models that represents the nonlinear dynamics of sound waves [41]. The dynamics of ultrasound generated from an array of transducers can be modeled with Burger's equation:

$$\frac{\partial p}{\partial x} = \frac{\beta}{\rho_0 c_0^3} \frac{\partial}{\partial t'} p^2 + \frac{\delta}{2c_0^3} \frac{\partial^2 p}{\partial t'^2}, \tag{2.2}$$

where β is the coefficient of nonlinearity, ρ_0 is the density of air, and c_0 is the sound speed. The first term on the right side has nonlinearity. By substituting Eq. 2.1 into p, we have

$$\begin{aligned} \frac{\partial}{\partial t'} p^2 = \frac{\partial}{\partial t'} [&p_c^2 \sin^2(2\pi f_c t') + p_{s\_}^2 \sin^2(2\pi f_{s\_} t') \\ &+ 2 p_c p_{s\_} \sin(2\pi f_c t') \sin(2\pi f_{s\_} t')], \end{aligned} \tag{2.3}$$

For simplicity, we calculate only the third term of Eq. 2.3, from which, we can derive f_m. [*1]

$$\begin{aligned}
&\frac{\partial}{\partial t'}(2p_c p_{s_} \sin(2\pi f_c t') \sin(2\pi f_{s_} t')) \\
&= 2[2\pi f_{s_} p_c p_{s_} \sin(2\pi f_c t') \cos(2\pi f_{s_} t') \\
&\quad +2\pi f_c p_c p_{s_} cos(2\pi f_c t') \sin(2\pi f_{s_} t')], \\
&= -2\pi p_c p_{s_}[(f_c + f_{s_}) \sin(2\pi(f_c + f_{s_})t') \\
&\quad +(f_c - f_{s_}) \sin(2\pi(f_c - f_{s_})t'), \\
&= -2\pi p_c p_{s_}[(f_c + f_{s_}) \sin(2\pi(f_c + f_{s_})t') \\
&\quad +f_m \sin(2\pi f_m t')], \qquad (2.4)
\end{aligned}$$

Eq. 2.4 contains two terms. The first term, which contains $\sin(2\pi(f_c + f_{s\_})t')$, will be removed by low–pass filter. Thus, remaining term is a sine function with the frequency of the original modulation wave, f_m. By substituting Eq. 2.4 into Eq. 2.2, we derive that $\partial p/\partial x$ contain the following term,

$$\frac{2\beta\pi p_c p_{s\_} f_m}{\rho_0 {c_0}^3} \sin(2\pi f_m t') \qquad (2.5)$$

By integrating the term with respect to x, we derive that p contains the following term

$$\frac{2\beta\pi p_c p_{s\_} f_m}{\rho_0 {c_0}^3} x \sin(2\pi f_m t') \qquad (2.6)$$

which indicates that the observed sound pressure includes the component of the original modulation wave. This is how the nonlinearity of the air demodulates the modulated sound wave.

Figure 2.3 presents an overview of the parametric phenomenon. After emitted from a parametric loudspeaker, the sound pressure of the audible sound wave, f_m, gradually increases. Although both the audible sound wave and inaudible ultrasound

[*1] If we compute the partial differentiation of the first and second terms in a way like Eq 2.4, sine functions with the frequencies of $2f_{s\_}$, $2f_c$, and so on, appear. Because these frequencies are not associated with f_m and will be removed by the low–pass filter on the microphone, all these sine functions can be omitted in the remaining calculation.

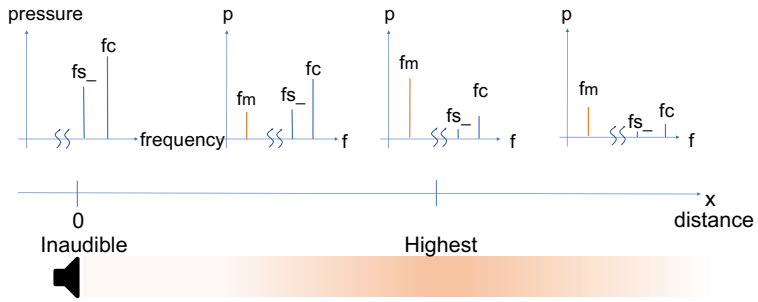


Fig. 2.3 Illustration of the demodulation in the air. f_c is a carrier frequency and $f_{s\_}$ is a sideband frequency, where $f_{s\_} = f_c - f_m$ and f_m represents a frequency of the sound wave to be injected by an attacker. In a short distance, the sound pressure of the demodulated sound, f_m will increase in proportion to the distance, x, following Eq. 2.6. However, due to the attenuation of the ultrasonic wave, the sound pressure of the demodulated sound will decrease over a long distance.

wave are to be attenuated over time, inaudible ultrasound waves attenuate faster due to the fact that in the air, high frequency sound wave attenuates faster compared to low frequency sound waves. The parametric phenomenon is observed only along the direction in which the ultrasound was emitted because the waves have the same phase along the path.

Finally, we show the intuitive explanation of the formation of directional sound beam. The demodulated sound traveling in the forward direction is amplified because the phase is aligned. On the other hand, sound traveling in a direction other than the forward direction is not amplified because the phase is not aligned. The mathematical description of the theory can be found in Refs [38, 39].

2.2.3 Voice Presentation Attack

In the ISO/IEC standard, presentation attacks are defined as "presentation to the biometric data capture subsystem with the goal of interfering with the operation of the biometric system. [42]" There have been several approaches for evading speaker recognition or, more broadly, voice authentication. These attacks are known as voice presentation attacks [4]. Well-known voice presentation attacks include the replay

attack [13, 43], speech synthesis attack [13], and voice conversion attack [14].

During a replay attack, an attacker pre-records the speech of the victim in advance. The attacker then replays the recorded speech to the target device. Distinguishing between genuine and replayed speech from the time-domain and spectrum-domain representations of speech data is difficult task [44]. The drawback of a replay attack is that an attacker needs to pre-record speech, including voice commands for both activation and recognition. *Speech synthesis* and *Voice conversion* are techniques that alleviate this limitation. Speech synthesis (Text-to-speech, TTS) is a technique to generate natural speech sound from the text. Wavenet [45] is one example that creates synthesized voices by using deep learning models. Voice conversion aims to convert an attacker's voice to a victim's voice in real time. We do not need to prepare text, unlike in TTS. These attacks offer an effective way to generate synthetic speech in a manner such that the generated output is perceived as a sentence uttered by a target. In [14], the author demonstrated that an attacker can successfully execute a voice impersonation attack by using an off-the-shelf voice-conversion tool, even against state-of-the-art voice verification systems. They reveal that the attacker can convert his/her voice if they collect just a few minutes' worth of audio.

While these attack techniques aim at impersonating the victim's voice, our goal focuses on the different attack vector, i.e., secretly delivering the voice signal to the target voice assistant device. As our attack is agnostic to the voice content, voice presentation attack techniques can be directly mounted on our attack.

2.3 Threat model and assumptions

In this section, we describe the Audio Hotspot Attack threat model by making several assumptions to evaluate the threat.

Target of the attack

The goal of an attacker is to manipulate the target voice assistance device without being noticed by people. Although the attack is applicable to various voice assistance systems in principle, a smart speaker is used herein as an example of the target device. Because smart speakers can control smart home devices, the attack vector ranges

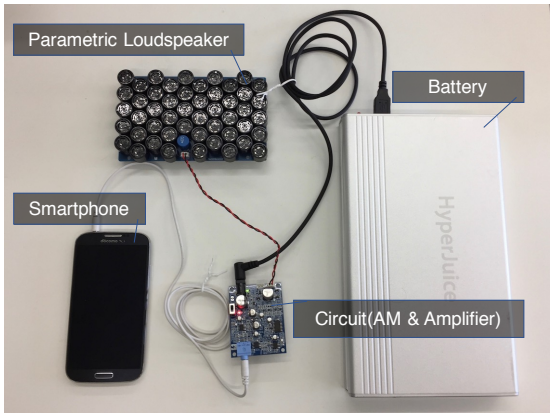


Fig. 2.4 An example of device setup. We use a battery to allow attackers to use this device anywhere. The circuit contains amplifier and amplified modulator. The details of the circuit are presented in Fig. 2.5.

are widespread. We evaluated the attack using two smart speakers, Amazon Echo and Google Home. For these devices, an attacker must activate the device with a wake-up word, and then transmit a voice command. In this study, we assume that the target device is not moving (i.e., it is set on a fixed place, for example, on the table). This assumption is natural in the case of smart speakers.

Attacker's equipment

As shown in Figure 2.1, the *Audio Hotspot Attack* has the two attack modes: *linear attack* and *cross attack*. An attacker needs to setup a parametric loudspeaker for the linear attack, and two parametric loudspeakers for the cross attack. The parametric loudspeaker that performs the attack is small and portable. The attacker also needs to carry a smartphone in order to generate malicious voice commands from the parametric loudspeakers. Figure 2.4 shows an example of a device setup used by an attacker to execute an attack.

Speaker recognition

As mentioned in Section 2.2, modern devices equipped with voice assistance systems such as smartphones or smart speakers have increasingly adopted the speaker recognition functionality. If the owner of a device has turned on this functionality, an attacker may not be able to succeed in the attack even when he/she has successfully

transmitted an inaudible voice command to the target device.

Here, the attacker collects voice samples by being in close physical proximity to the target, by making a phone call, or by searching for clips online. For the purposes of this work, we assumed that an attacker was able to bypass the speaker recognition by leveraging voice presentation attacks, which are discussed in Section 2.2.3. As shown in Section 2.7.2, there are some methods that detect presentation attacks (PAD method). We assume that the voice assistance systems do not have a PAD method. We confirm that presentation attacks are successful on practical devices, i.e., Google Home and Amazon Echo, before the experiments.

2.4 Experimental setup

In this section, we describe the design of our experiments, including details pertaining to the devices, equipment, and software used, together with their settings.

2.4.1 Materials

Experiment room

Sound wave dynamics depend on the material makeup of the room. As these attacks were performed using sound waves, the choice of the experiment room was key. Otherwise, the obtained results will be valid only for a specific environment. To overcome this concern, we used a room designed for acoustic experimentation. To eliminate the effects of the material makeup of the room, all wall and ceiling surfaces were made of sound-absorbing material (Appendix B, Figure 2).

The average sound pressure level (SPL) of the room was around 12 dB(A). Here, dB(A) denotes A-weighted SPL, which is applied to instrument-measured sound levels. A-weighting is used because the human ear is less sensitive to lower audible frequencies.

Target devices

Following the assumption that the target device is stationary, Google Home and Amazon Alexa are the primary target devices used for the analysis. These devices

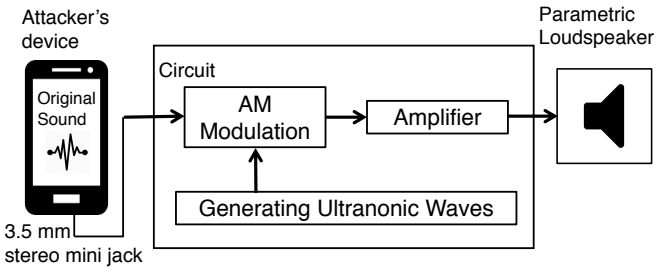


Fig. 2.5 Circuit diagram. The circuit first applies AM to the input soundwave, using the generated ultrasonic wave as a carrier wave. Next, the sound pressure of the AM wave will be amplified. The amplified soundwave will be the output for the parametric loudspeaker.

were chosen because they accounted for more than 95% of the smart speaker market share in 2018 [46].

Equipment used for the experiments

Table 2.1 shows a list of equipment used for the experiments. While there are several commercial parametric loudspeaker products, we intended to take a white-box approach. That is, as the details of the board and elements are publicly available on the manufacturers' websites, we can obtain the technical specifications of the speaker, such as frequency response. To this end, the Switch Science Super directional speaker [47] was adopted as a primary parametric loudspeaker. The kit comprises two printed circuit boards (PCBs). One PCB has an AM circuit, an amplifier circuit, an audio input (3.5 mm stereo mini jack), and a power input (DC 12V/1A). Figure 2.5 presents a diagram of the circuits. Another PCB implements 49 ultrasonic ceramic transducers connected in parallel. The first PCB applies the AM to the input sound wave and then amplifies the signal level. The amplified signal is transmitted to the second PCB, i.e., ultrasound transducers. Another parametric loudspeaker—directional speaker ACOUSPADE—is also used, to study the maximum distance at which the attack can succeed. The sound level meter is capable of measuring the SPL of 28–138 dB(A) for a frequency range of 20 Hz to 20 kHz. The meter was used to measure the SPL of several areas in the experiment room

Table 2.1 A list of equipment used for the experiments.

Equipment	manufacturer / model number
Parametric loudspeaker	Switch Science / SSCI-018425 [47]
Amplifier	Accuphase / Power Amplifier PRO-15 [48]
Parametric loudspeaker	Ultrasonic audio technologies / Directional Speaker Acouspade [49]
Dynamic loudspeaker	YAMAHA / MONITOR SPEAKER MS101 III [50]
Sound level meter	RION / NL-32 [51]
Ultrasonic microphone	B&K / 4939-A-011 [52]
Audio Interface	MOTU / UltraLite mk4 [53]

under various conditions. The ultrasonic microphone was also used for measuring the ultrasonic components in the measured sound waves.

2.4.2 Voice generation

To generate a malicious voice speech command, we used Amazon Polly [54], a cloud service that turns text into natural sounding speech. As the basis for the analysis, the voice named "Ivy" was used, which is a female, US English accent. The voice parameters (e.g., speaking rate or fundamental frequency) were set to default values. All voice assistance systems that were tested to check whether they accept synthesized voice commands. As speech synthesis services can change in the future, we plan to make our data available to any researchers who wish to replicate or extend our work.

2.5 Evaluation of the attack

We evaluated attack feasibility using the following aspects: maximum successful attack distance, noise tolerance of the attack, and the impact of voice commands.

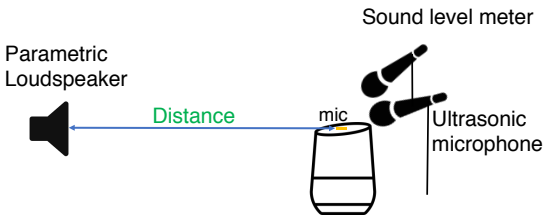


Fig. 2.6 Experimental setup of distance measurement experiments. The distance measured was between the parametric loudspeaker and the microphone of the voice assistance systems.

For simplicity, and to evaluate the impact of these factors, we applied a linear attack. For the cross attack, we evaluated attack feasibility using the parameters obtained through the linear attack experiments. The attack success depends on the type of voice command (i.e., activation or recognition). Therefore, for each attack mode, we applied both types of voice commands. In general, activation commands ("wake-up words") are more likely to succeed.

2.5.1 Distance versus Attack success rate

The aim of this study was to clarify how the distance between the target device and adversary's parametric loudspeaker affected the success rate of the Audio Hotspot Attack. Throughout the experiments, the SPL of the output power from the parametric loudspeaker was fixed. In particular, the audible sound of the parametric loudspeakers was adjusted to 60 dB(A), and the SPL of the ultrasound was 100 dB at a point 3 m away from the parametric loudspeaker. Figure 2.6 presents the experimental setup. The distance measured was between the parametric loudspeaker and the microphone of the voice assistance systems.

To measure the distance, we used the experiment room (described in section 2.4.1). We extended the study to three different locations, including a hallway, seminar-room, and outdoors.

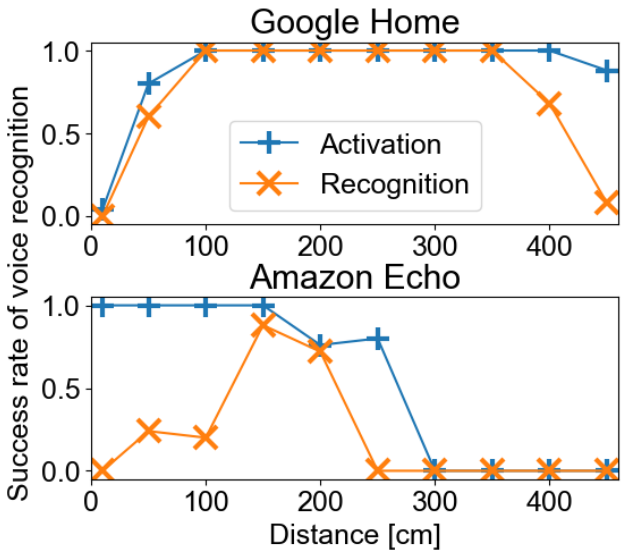


Fig. 2.7 Distance versus attack success rate. Noise SPL is set to 60 dB(A). For Google Home, the longest distance was 3.5 m. Activation voice commands were more likely to be accepted compared to recognition voice commands.

Measurement within the experiment room

The distance between the target device and the parametric loudspeaker was altered from 0.1 m to 5 m in increments of 50 cm (i.e., 0.1, 0.5, 1.0,..., 5.0 m). By adjusting the output power of the dynamic speakers, we were able to adjust the SPL of the noise measured in the room to 60 dB(A) with error bounds within 1 dB(A). Notably, a SPL of 60 dB(A) corresponds to an environment where a person's speech is heard at a distance of 1 m. Thus, the noise level was fairly high. This setting was purposively chosen to conservatively evaluate attack success rate (i.e., a higher attack success rate could be expected in quieter settings). We note that the 1/f noises better suited to emulate a realistic environment than the white noise because it is natural that signals with the lower or higher frequencies have more or less power respectively.

For a given distance, a pair of activation and recognition voice commands were generated. This process was repeated 25 times. For each voice command, we noted if the command was accepted by the voice assistance system by observing the response

of the device. For the activation commands, "*Ok Google*" for Google Home and "*Alexa*" for Amazon Echo were used. For the recognition voice commands, "*What's on my next schedule*?" for Google Home and "*What's on my schedule*?" for Amazon Echo*[2] were used.

The attack success rates were calculated, and the results are shown in Figure 2.7. For a certain range of distances, the attack was highly successful for both devices. This was particularly true for Google Home, the longest distance was 3.5 m. Activation voice commands were more likely to be accepted than recognition voice commands. This makes sense given the fact that the recognition voice commands are much more variable than activation voice commands. In the short distance, the success rate becomes low because the acoustic sound was too loud to be properly processed by the voice assistance systems. Finally, Google Home featured a higher attack success rate than Amazon Echo. As these commercial products are black box in nature, their behaviors can be difficult to interpret. It is possible that circuits and software used for Amazon Echo are somehow resistant to the Audio Hotspot Attack; therefore, they will be investigated in future studies.

Extended measurement in practical environments.

Next, we studied the distances of successful attacks using different locations: a hallway, a seminar room, and outside. The hallway and the room have much higher reverberation compared to the room dedicated for acoustic experiments. We used a commercial parametric loudspeaker product [49], as listed in Table 2.1. The parametric loudspeaker can emit full frequency-range speech with the audible SPL of 62–63 dB(A) at a distance of 3 m. For reference, the location photos are shown in Appendix B. Note that for these locations, we did not add synthesized noise sounds. The average SPL measured in the hallway was 39.3 dB(A), the seminar room was 55.2 dB(A), and the average outside SPL was 52.5 dB(A). The conditions outside were as follows: the weather on the day was fine, with temperature was 23.2 °C (73.8 °F), a humidity of 36%, and a wind speed of 6 m/s southward. Note that, we do not use synthesized noise in this measurement, to evaluate the effect of noise on

*[2] At the time of the experiment, Alexa did not support the 'next' voice command for the calendar.

Table 2.2 The longest distance the attack was effective at a hallway, a seminar room, and outside. In the hallway experiment, the attack was effective at a distance of 10+ m. In the case of the hallway and room, the longest distance is 4+ m. We show the picture of each place in Appendix.

	Hallway [m]		Room [m]		Outside [m]	
Devices	Acti.	Recog.	Acti.	Recog.	Acti.	Recog.
Google Home	15.0	11.7	4.2	4.0	4.2	4.2
Amazon Echo	19.9	12.1	4.8	4.0	5.8	4.2

realistic environments. The purpose of the experiment was to determine the longest distance at which the attack is still effective, with the effectiveness being determined using the following criteria: if three consecutive voice commands are all accepted for a given distance, the attack is regarded as effective for the distance. For each location, the starting distance was 1 m and the tests were repeated until there was an attack failure. Tables 2.2 summarize our results.

The hallway experiment demonstrated that the attack was effective at a distance of 10+ m. The seminar room and outside experiment demonstrated that the attack was effective to a distance of 4+ m. The difference in the attack success distances reflects the respective noise levels within each location. These results indicated that the Audio Hotspot Attack was feasible in three real-world scenarios. We can succeed in the attack in two environments with reverberation, i.e., the hallway and inside the room. We also showed that the experiment was successful outside the room. In addition, the attack success distances achieved were much longer than the state-of-the-art inaudible voice command attack that uses ultrasound [8], which indicated that the maximum distance for Amazon Echo averaged 1.65 m with a background noise of 55 dB SPL.

2.5.2 Noise tolerance

We studied how the noise affects the attack success rate. For this study, we used the experiment room, as described in Section 2.4.1. Because we were examining the effects of noise, the sound generated by the parametric loudspeaker was fixed at 60 dB(A) and the distance between the parametric loudspeaker and the target device was 1.5 m.

Stationary noise

Using the dynamic speaker, we generated $1/f$ noise with an SPL ranging from 45 dB(A) to 78 dB(A) (the maximum SPL for the dynamic speaker). the common environmental noise levels are shown in [55]. To calculate the signal-to-noise ratio (SNR), we use the following formula Eq. 2.7 [56]

$$\text{SNR dB} = \text{SPL of sound dB} - \text{SPL of noise dB} \tag{2.7}$$

We use the sound level meter to measure the SPL of voice command and noise. Figure 2.8 shows the results. For both devices, the attack was most successful when the noise SNR was over than 0 dB, i.e., when the input command and noise have the same volumes. Activation voice commands were more tolerant of noise. This observation agrees with those previously-described in Section 2.5.1.

Nonstationary noise

We evaluate noise tolerance in an environment that has nonstationary noise. As nonstationary noise, we adopt babble noise. We used the room dedicated for acoustic experiments. We chose three types of noise settings: Default, Speech Blocker, and Chic dinner, which are taken from Ref. [57]. These noise types contain conversations in English. We summarize the results in Figure 2.9. We attempt to input the voice command 10 times in each setup. For both devices, the attack was successful when SNR was -5 dB and over. When the SNR is 0 dB, i.e., when the volumes of input command and noise are same, attacks sometimes failed. In other cases, these results follow the observation of Fig 2.8.

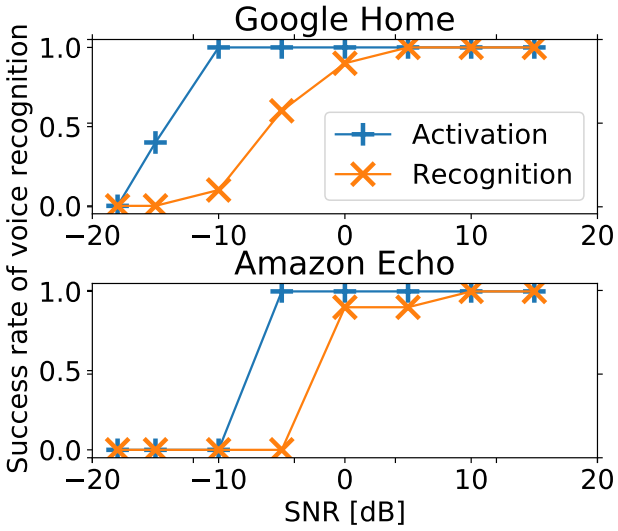


Fig. 2.8 Stationary noise versus attack success rate. The audible sound from the parametric loudspeaker was fixed to 60 dB(A). The attack was most successful when the SNR was larger than 0 dB.

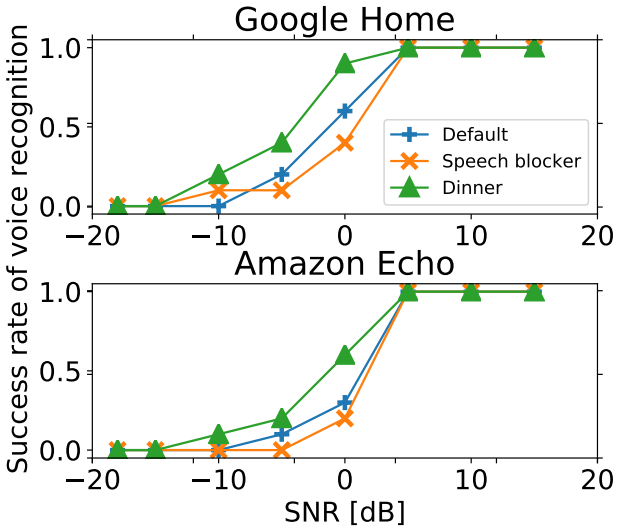


Fig. 2.9 Non stationary noise versus attack success rate. We used the recognition command for each device. These results follow the observation of Fig 2.8.

2.5.3 Impact of voice commands

To study the impact of voice commands, various commands are inputted into the target devices. In this experiment, the distance between the parametric loudspeaker and the target devices was fixed at 1.5 m. Again, the output audible SPL of the parametric loudspeaker was set to 60 dB(A). Each command was tested 10 times.

Table 2.3 shows the results. As indicates by the results, the attack success rate was high for commands of short lengths. We note that although the lengths of these commands were short, they can be used for malicious purposes; for example, by starting with the recognition command "Set volume 0," an attacker can improve the probability of success for the next attacks as a voice response from the device will not be heard by a nearby person. The attacker can also turn IoT devices on/off. If this device is a piece of heating equipment, considerable physical damage is possible. In contrast, for longer commands, the attack success rate was low.

We conjecture that there are several reasons behind this observation, e.g., the occurrences of infrequent words or the accumulation of recognition errors. These results agree with [8], who showed that longer commands, emitted as ultrasounds, were prone to failure.

2.5.4 Evaluation of the cross attack

To perform the cross attack, the AM sound wave was separated into the carrier wave and the lower sideband wave using MATLAB [58]. The two sound waves were amplified and emitted through the two parametric loudspeakers. The amplifiers were adjusted so that the SPL of the audible sound was at its maximum at the target area (center of the room). The average SPL of audible sound was 42.7 dB(A). The cross attack was tested by changing the position of the target device, as shown in Figure 2.10 (Right). In the figure, the blue circles indicate measurement points, where a sound level meter was set. Two parametric loudspeakers were set so that they would cross at the center point. Unlike the linear attack setup, this setup was not symmetrical and each parametric loudspeaker transmitted a different signal (i.e.,

Table 2.3 Attack success rates for various voice commands. The attack success rate was high for commands of short length (2–5 words.) The commands "turn on / off [device name]" are used for many smart home devices. The commands "turn in to 0" or "Set volume 0" change the volume minimum, which can make the output of device stealthy.

Device	Voice commands	Success rate
Google	OK Google	10/10
	Max volume	10/10
	Turn in to 0	10/10
	What's on my next schedule	10/10
	Turn on the light	10/10
	Turn off the light	10/10
	Play some music	10/10
	Tell everyone my password is abc	5/10
	Broadcast my credit card number is 1234567890	3/10
Amazon	Alexa	10/10
	Pair devices	10/10
	play some music	10/10
	What's on my next schedule	9/10
	Set volume 0	9/10
	Turn on the light	9/10
	Turn off the light	10/10
	Tell everyone my password is abc	2/10
	Broadcast my credit card number is 1234567890	1/10

a carrier wave and a sideband wave, respectively). We established $5 \times 5 = 25$ measurement points. As shown in the figure, we installed four dynamic speakers to fine-tune the SPL of ambient room noise. We configured the directions of the dynamic speakers such that noises were equally distributed throughout the room. We fixed the distance between the target device and two parametric loudspeakers to

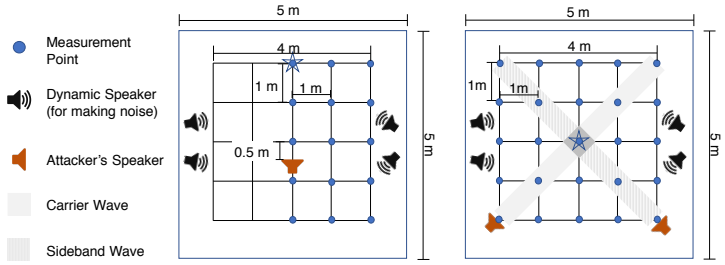


Fig. 2.10 Overview of the experimental setup. Left: user study of the linear attack in the acoustic room. Right: user study of the cross attack in the acoustic room. We use four dynamic speakers to adjust the noise level.

$2\sqrt{2}$ m, and the SPL of noise was set to 43 dB(A).

At each position, the attack was repeated 10 times, with the number of successes counted. Figure 2.11 shows the results. The first finding was that the attack was successful only in the area targeted by the cross attack. Second, for the activation voice command, the attack success was 100% for both devices. Finally, although the success rate was low for voice recognition ("what's on my next schedule?"), it remains a realistic threat, given the fact that an adversary can repeat the attack until it succeeds.

2.5.5 Summary

Throughout this section, we evaluated attack feasibility. First, the experiments demonstrated that the attacks were successful over long distances. In the experiment room (500 cm $\times$ 500 cm), Google Home attacks were 100% successful at 350 cm and Amazon Echo attacks were more than 90% successful at 150 cm. The hallway experiments demonstrated that, for both devices, attacks were successful at distances greater than 10 m. Second, we discovered that the attacks were tolerant of environmental noise. For both devices, the attack success rate remained high at a noise SPL of 60 dB(A). This SPL corresponded to the SPL used for the experiments described in Section 2.6. Finally, the attacks were successful with various types and lengths of voice commands.

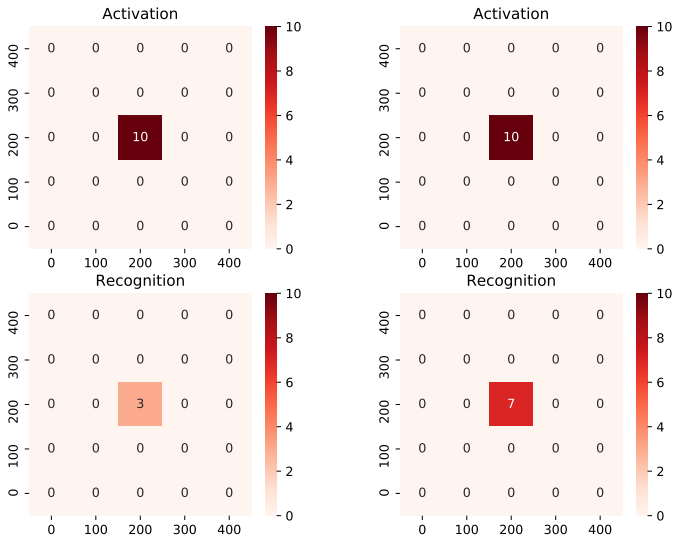


Fig. 2.11 Number of successful cross attacks at each position (max is 10). Top: Activation and Bottom: Recognition. Left: Google Home, and Right: Amazon Echo. The demodulation point was adjusted to the center, point (200, 200).

2.6 Human study experiments

In psychoacoustics, hearing is different from objective SPL measurements [59]. We tested to confirm whether the directional sound generated from parametric loudspeakers could be perceived by humans around the targeted device. To this end, we conducted extensive user study experiments to answer the RQ3: "Is the attack stealthy for nearby people and unrecognizable for them?" To complement the results of our human studies (subjective evaluation), SPL measurements were taken with the sound level meter (objective evaluation).

2.6.1 Experimental setups

Figure 2.10 presents an overview of the experimental setup. For the linear attack mode, both a parametric loudspeaker and a dynamic speaker were used to observe their differences. In the figure, the blue circles indicate measurement points, where a participant was seated. As the setup was symmetric in nature, $3 \times 5 = 15$ measurement points were set only in the right half. We omitted the left half to reduce the workload of the participants without sacrificing the generality of the results. The distance between the measurement points was set to 1 m. For the cross attack, two parametric loudspeakers were set so that they would cross at the center point. We established $5 \times 5 = 25$ measurement points, with a chair at each measurement point (See Appendix B, Figure 2).

The output power of the adversary' s parametric / dynamic loudspeakers was adjusted so that the SPL of the audible sound (not the ultrasound) measured 3 m away from the parametric loudspeaker was 60 dB(A). Accounting for the inaudible sound wave, the total SPL was 120—130 dB(A) for all these settings. Finally, for the four dynamic speakers that generate $1/f$ noise, we adjusted the output power such that the audible SPL was 60 dB(A) at a distance of 3 m. For reference, the SPLs of common environmental noises are summarized in [55].

2.6.2 Human study overview

Participants

For the user study, we recruited 20 normal-hearing participants. Of these, 12 were female and eight were male, with ages ranging from 19 to 27. Thus, the participants were younger on average. Because younger people tend to have better hearing, we selected a severe condition to evaluate recognizability.

The participants consist of students at our university. We let the participants choose the preferred language from the two choices, Japanese and English. While 16 participants who selected Japanese are all native speakers of Japanese, other three participants who selected English were fluent in English but not necessary were the

native speakers of English. Two of them are from Indonesia and the other is from China. For each participant, consent was obtained before enrolment. All participants were informed that they could quit the experiment whenever they desired. Other ethical considerations are discussed in section 2.7.

Procedure

For each setup, each participant was first directed to sit in a chair set at the position marked with the star symbol in Figure 2.10. Then, the height of loudspeaker(s) was adjusted so that the participant's sitting height matched the position of the loudspeaker(s). For each participant, the heights and angles of the speakers were fixed throughout the experiments. After the beginning of a session, a random word is emitted twice from the speaker at a random moment in time. A participant reports whether they recognize the word. If they recognize it, they write down the word that they recognized.

From the set of random words, those containing between 3 and 6 phonemes were selected. It was also ensured that the words would be difficult to predict beforehand, e.g., wake-up words typically used for voice assistance systems were avoided. Each participant repeated the sessions after moving to another chair.

To ensure the quality of the subjective evaluations, we used a silent task with each participant. During the silent task, no voice sounds were emitted. If a participant reported that they heard something during the silent task, the other results reported by the participant were considered unreliable and removed. Consequently, two participants' results were removed from the final analyses.

Evaluation of recognizability

To quantify the recognizability reported by the participants, we used a Jaccard index for the sets of letters in two words t and r, which are a test word and a reported word, respectively. For instance, if a test speech word is 'fest' and the reported word is 'test', the Jaccard index is computed as $J(\text{'fest'}, \text{'test'}) = 3/5 = 0.6$. For reference, a randomly sampled answer sheet reported by one of the participants is shown in Appendix B.

In total, for each measurement point, we collected 18 scores reported by the 18

participants. At least one score for each measurement point was In total, for each measurement point, we collected 18 scores , reported by the 18 participants. At least one score for each measurement point was omitted, as there was one silent task for each participant. To quantify the recognizability, the average of the reported scores was taken for each measurement point.

2.6.3 Results of the human study

Figure 2.12 shows the linear attack results. The heat maps represent the average Jaccard index scores. Notably, for the dynamic loudspeaker experiment, most participants successfully recognized the test speech words across a wide range. In fact, the test words were audible even behind the speaker. On the other hand, for the parametric loudspeaker experiment, the audible space was limited to a narrow area (i.e., the direction of directional sound propagation). The generated sound wave was somewhat inaudible over a short range owing to the fact that the generated ultrasonic beam moved forward before it was demodulated in the air.

Figure 2.13 shows the cross attack results. It is important to note that there seem to be no audible spaces in the room. However, as shown in the previous subsection 2.5.4, the cross attack was successful in emitting malicious voice commands to the voice assistance systems. This contradiction can be explained as follows: as the cross point was limited to a very narrow area, it did not “hear” the areas close to the participant's ears. Even if a participant was able to catch either a carrier wave or a lower sideband wave, they would not recognize them unless they caught both sound waves at a cross point. To complement the results of the human study, the results of the objective sound level meter evaluations are presented in Figure 2.14.

2.6.4 Summary

In this section, we examined the recognizability of sounds generated from parametric loudspeakers. For comparison, we also examined the characteristics of the sound generated by a dynamic speaker. Both the subjective and objective evaluations revealed that the directional sound generated from the parametric loudspeakers

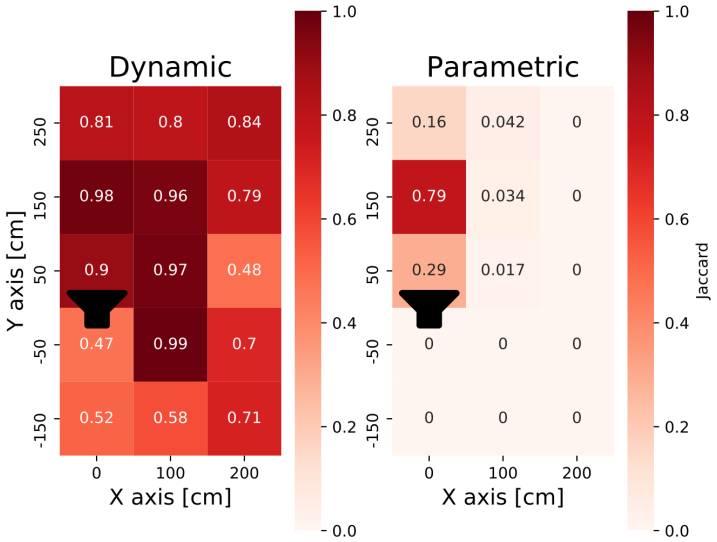


Fig. 2.12 Average Jaccard index scores of the linear attack measured in a 200 cm × 400 cm area. Left: dynamic speaker and Right: parametric loudspeaker. The point (0, 0) is defined as the location of the loudspeaker. User cannot hear the on space except in front of the parametric loudspeaker.

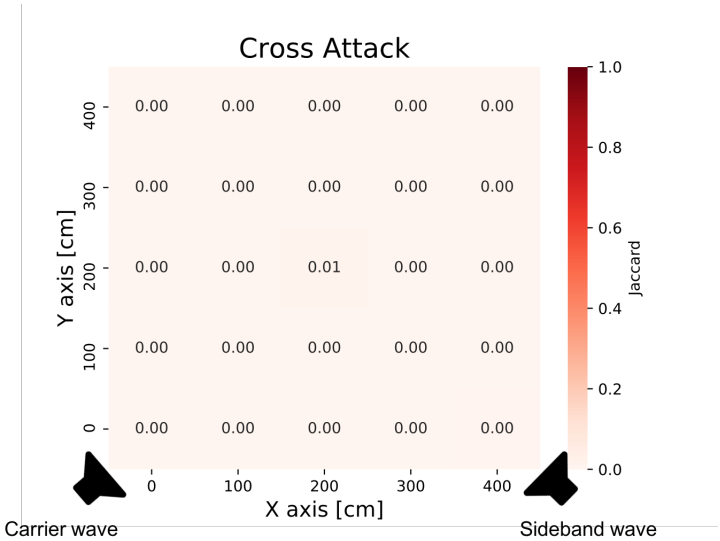


Fig. 2.13 Average Jaccard index scores for the cross attack measured in a of 400 cm × 400 cm area. The point (200, 200) is defined as the demodulation point. We found that the users cannot hear sound waves everywhere except in the center.

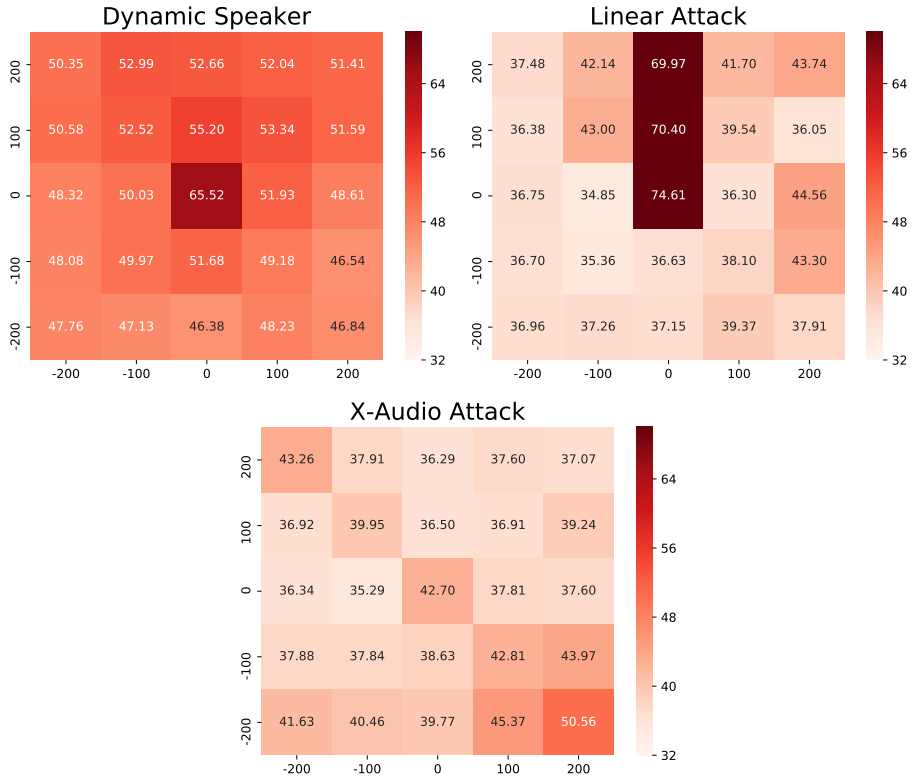


Fig. 2.14 SPL measured for the three attack modes. The unit for the numerical values is dB(A). The setup is same as in the human study. We have the speaker on the point (0,0) in the case of the dynamic speaker and linear attack. In the case of the X-Audio attack, (0, 0) is the demodulation point for voice commands.

achieved sufficient unrecognizability to perform the Audio Hotspot Attack. Specifically, the sound generated with the cross attack was difficult for a human near the target device to perceive.

2.7 Discussion

In this section, we discuss the limitations and extensions of Audio Hotspot Attack, possible countermeasures against it, and ethical issues considered during the experiments.

2.7.1 Limitations and possible extensions

Because the Audio Hotspot Attack uses sound wave(s) to inject malicious voice commands, it will not succeed if there is an obstacle between the target device and the parametric loudspeaker(s) (e.g., a wall or a window). This limitation also applies to other inaudible voice command attacks [9, 60, 61]. One possible method of overcoming this limitation would be to install parametric loudspeaker(s) on a ceiling, thus creating a "sound shower." In fact, parametric loudspeakers are often mounted on ceilings to make sounds audible only at one point in the room, without the risk of interruption form an obstacle. Even when it is unrealistic to mount a parametric loudspeaker on the ceiling, it would still be effective to place it at a raised or a side position to ensure that the sound wave emitted avoids obstacles.

We used two smart speakers, Google Home and Amazon Echo, as examples of popular devices with voice assistance systems. Other types of voice assistance systems include smartphones, in-car navigation systems, and commercially available medical devices. Studying the effectiveness of the Audio Hotspot Attack on most of these other devices will be conducted in future studies; however, we did verify that the attack worked on several smartphones. Although the evaluation of the latter is not as thorough as that presented in section 2.5, some results have been given in the Appendix for reference.

Finally, although we sought to make these studies scientifically reproducible, the target devices are updated regularly. Furthermore, as the majority of the off-the-shelf voice assistant devices today run the speech recognition on the server side, it is prone to change over time. Therefore, once changes are made to the hardware or software in the voice assistance devices, other results may differ from the ones we obtained. As off-the-shelf products are "black box" in nature, it is difficult to fully understand how input sound waves are processed by the device's hardware and/or software. Therefore, to make the results of the experiments to be invariable and reproducible, it would be desirable to develop open-source hardware and software platforms, which would allow researchers to share and compare results using similar tools. At present, we are developing such a platform so that interested researchers

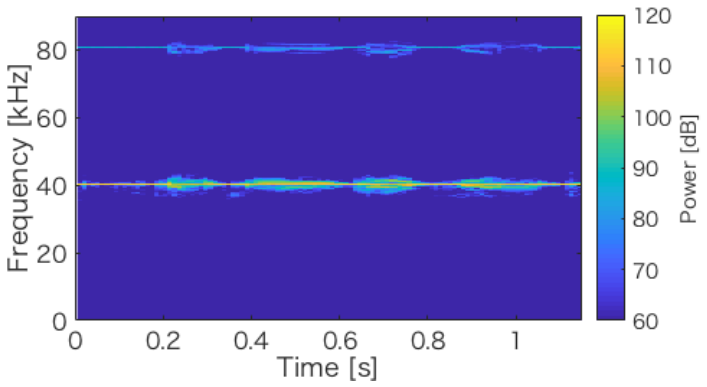


Fig. 2.15 Spectrogram of a speech signal emitted from a parametric loudspeaker. The signal was recorded with an ultrasonic microphone. The frequency range was set above 20 kHz (inaudible frequency). The content is "OK Google".

can conduct further work on security and privacy issues related to voice assistance systems.

2.7.2 Countermeasures

Audio Hotspot Attack leverages the natural phenomenon of ultrasound self-demodulation in the air; therefore, it is not practical to try to block voice commands before they reach the target device. One possible solution is to detect the voice commands and differentiate them from others that are legitimate. There are two ways to achieve this goal. An easy and effective approach is to employ speaker recognition; in fact, smart speakers such as Google Home or Amazon Echo have already adopted this functionality. However, as discussed in Section 2.3, such approaches are still vulnerable to advanced replay or voice-morphing attacks. Therefore, we require methods that can detect voice commands being emitted from parametric loudspeakers. In the following section, we discuss three potential approaches to achieve this goal.

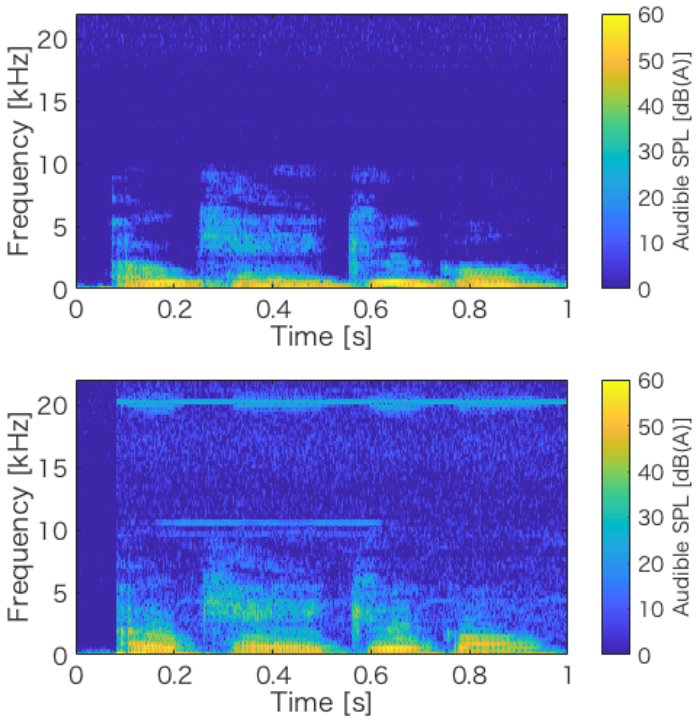


Fig. 2.16 Spectrogram of a speech signal emitted from a dynamic loudspeaker (top) and a parametric loudspeaker (bottom). The signals were recorded with a normal microphone. The frequency range was set below 20 kHz (audible frequency). We can see the folding noise at 10 kHz and 20 kHz in the bottom spectrogram. The content is "OK Google".

Detecting ultrasonic sounds

Although the ultrasounds emitted from a parametric loudspeaker are demodulated in air, there are un-demodulated ultrasonic components in the observed sound wave. Figure 2.15 shows the spectrogram of a speech signal emitted from a parametric loudspeaker. The original speech data was "Ok Google," which was generated using Amazon Polly (Ivy). In the spectrogram, the power of the ultrasonic component is around 40 kHz, which corresponds to the carrier frequency of the AM-modulated sound. A harmonic overtone around 80 kHz was also observed. Thus, even ultrasound is self-demodulated in the air, and it is possible to observe ultrasonic

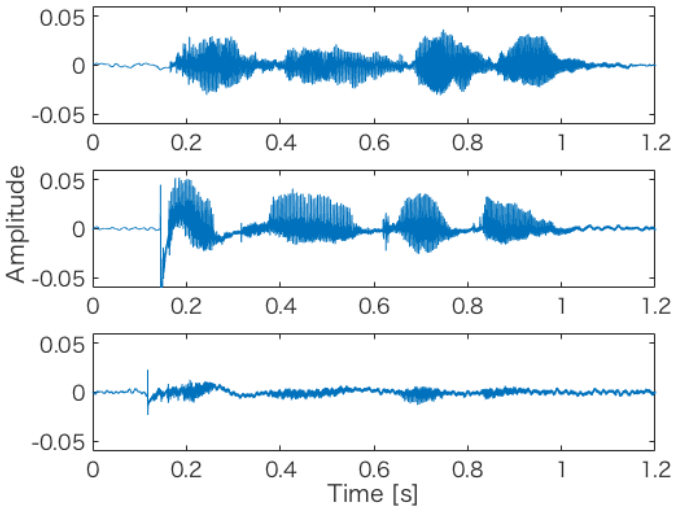


Fig. 2.17 Speech signals generated from a dynamic loudspeaker (top), a parametric loudspeaker (middle, linear attack), and Bottom: two parametric loudspeakers (bottom, cross attack). The content is "OK Google".

components of sound waves.

A straightforward approach to detecting such ultrasonic components is to apply an ultrasonic sensor. Although ultrasonic microphones are expensive, ultrasonic sensors are cheap and readily available. As Zhang et al. suggested [60], using MEMS microphones on mobile devices could be an alternative solution, as these microphones can sense acoustic sounds with frequencies higher than 20 kHz. Once a device detects the non-negligible amounts of ultrasonic components of a received sound wave, it may suspend the operation and require interaction with the device owner to resume the operation.

Analyzing the frequency patterns of audible sounds

Figure 2.16 presents the spectrograms of a voice signal ("OK Google" as spoken by Amazon Polly) emitted from a dynamic loudspeaker and a parametric loudspeaker. Although the original voice data was the same, there are different characteristics in the frequency patterns of the observed sound waves. As can be seen in Eq. 2.4 (Section 2.2), the SPL of the sound wave generated from a parametric loudspeaker

is proportional to the frequency of the original sound signal. This indicates that if the sound is emitted from a parametric loudspeaker, higher or lower frequency components are more or less likely to be observed, respectively, at the target. The horizontal lines shown in the lower spectrogram correspond to the *folding noise*, which is also known as *aliasing*. We can detect attacks if we observe the folding noise in spectrograms. To validate the effectiveness of this approach, we performed a brief experiment. From a given sound wave, we extracted components that had the frequencies above 10 kHz, which is over the audible frequency of 8 kHz. We then computed the power of the extracted sound wave. While the normal sound wave had almost zero power, the sound wave of the directional sound beam had non-zero power. By simply applying a threshold-based detection, we were able to distinguish the sound emitted by a loudspeaker from the one emitted by a parametric speaker with 100% accuracy.

Figure 2.17 shows speech signals emitted from a dynamic loudspeaker and parametric loudspeakers. Again, these speech signals were generated by the same original voice signal ("OK Google"), via Amazon Polly (Ivy). For the speech signals emitted from parametric loudspeakers (middle and lower panels in the figure), there is an intrinsic spike at the beginning of the speech signal. These spikes can be used as a fingerprint for detecting speech generated from a parametric loudspeaker. These spikes and other intrinsic characteristics can be used to differentiate speech generated from a parametric loudspeaker compared to speech generated from a regular voice using heuristics or machine learning-based approaches.

Voice Presentation Attack Detection (PAD) method

As inaudible voice command attacks will be combined with the presentation attacks, we can apply the presentation attack detection (PAD) method, which we assumed our target voice assistant systems had not implemented, to detect an Audio Hotspot Attack [4]. The ultimate countermeasure against such an attack is to be able to distinguish a synthesized voice from an authentic human voice. Liveness detection [62, 60, 63], which judges whether an input voice has come from a human or a dynamic speaker, is an example of the PAD method that could achieve this

goal. In real environments, attacks on speech recognition devices are by means of the latter. Therefore, it would be sufficient for a voice assistance system to be able to judge whether a sound comes from a human or a dynamic speaker, even if it is unable to identify a specific individual. Voice Gesture [60], as proposed by Zhang et al., attempts to detect the movement of a person' s mouth, by using changes in ultrasonic waves that occur as a consequence of the mouth movements and the position of the tongue when an approximately 20 kHz ultrasonic wave is emitted from a smart device (e.g., a smartphone or tablet) to the mouth of the user. This method detects differences in movement between a mouth and a dynamic speaker. The mouth movement changes for each pronunciation variation, whereas the surface of a dynamic speaker exhibits very little movement. The liveness detection method could be used to detect an Audio Hotspot Attack because ultrasonic transducers use fewer movements than the human mouth.

In our experiments, we have shown that simple rule-based or threshold-based detection work as countermeasures against the Audio Hotspot Attack. However, more robust countermeasures will be required in realistic environments. In [4], some typical countermeasure methods using the machine learning model are proposed. On the contrary, in [31], the authors pointed out that the machine-learning model does not work well for the datasets obtained in different setups. Overcoming the problem of overfitting to the specific datasets and/or environments is left for future work.

2.7.3 Ethical Considerations

Human study research

We performed a human study to test the unrecognizability of the Audio Hotspot Attack using parametric loudspeakers. The experiments were carefully designed such that they did not impose a burden on either the hearing or psychological states of the participants. The procedure for the human study was approved by the ethical review board at Waseda University. Prior to the experiments, we performed a pilot study to ensure the validity of our measures. Then, Participants were provided with all information required to make a meaningful decision as to whether or not they

were willing to participate in the experiment (informed consent). We explained the reasons for conducting the study, what the experimental procedures, potential risks and benefits were, and the ways in which participants could get more information on the study. The SPL of the sound waves was sufficiently low such that it did not cause the participants any discomfort. Participants were also given two-minute breaks every ten minutes and were able to stop participating at any time without incurring any penalty.

Offensive security research

The objective of this work was to explore the feasibility of the threats caused by inaudible voice command attacks. It was demonstrated that inaudible voice command attacks are viable through methods such as an Audio Hotspot Attack. Although this attack was proof of concept, we have also provided potential countermeasures by which they can be counteracted. Furthermore, with the aid of the national CERT, we have initiated communication regarding this with several manufacturers of voice assistance systems. Feedback, including plans for implementing the countermeasures within the products concerned, has been received. By the time of publication, vendor reaction will have been received and will also be reportable.

2.8 Related Works

Voice command attacks

DolphinAttack [8, 61] is an attack that inputs inaudible commands on a target microphone by AM modulating the sound, with the ultrasound as the carrier wave. The basic idea is based on the fact that the output of the MEMS and ECM microphones that are mounted on smartphones has nonlinearity [32, 9]. A nonlinear term is obtained by squaring the input signal in the output signal when an AM ultrasonic signal by the prepared voice is inputted to the microphone. That is, the output of the microphone receiving the AM-modulated ultrasound includes the frequency component of the original speech signal, and the speech recognition algorithm of the system that received the low-pass filtered signal is applied as recognized speech,

even though the input signal only generates high-frequency waves. The output generated by the nonlinear term has a smaller voltage value than the normal output and therefore it is easy to detect.

On the other hand, in an Audio Hotspot Attack, there is a marked difference in that *audible* sounds, which have been self-demodulated from the ultrasound waves, are received by a target device. This phenomenon is established because air is nonlinear and demodulates the AM-modulated ultrasonic signal, as shown in Section 2.2. Indeed, we cannot eliminate nonlinearity from the air because it is a natural phenomenon. In other words, even if microphone nonlinearity is completely removed, Audio Hotspot Attacks are still feasible even though inaudible voice commands are infeasible. In addition, Audio Hotspot Attacks can be employed from greater distances than DolphinAttacks because ultrasound has higher-than-audible frequencies, and therefore, it decays faster.

Audio adversarial examples

Audio Adversarial Examples [64] apply Image Adversarial Example [65, 66] techniques to voice waves. Adversarial examples are input to machine learning models that an attacker has intentionally designed to cause the model to make a mistake. The recognition results of the machine learning model are easily affected by a small amount of perturbation (small noise). Adding a small amount of noise to the original sound intentionally results in erroneous recognition. Therefore, Audio Adversarial Examples can be misidentified as arbitrary commands. The user cannot notice the subtle additional noise and targeted malicious commands are therefore executed on the voice assistant.

Existing attacks assume that software or hardware vulnerabilities are related to attack successes. Hidden voice commands and Audio adversarial examples use the vulnerabilities inherent to machine learning, and DolphinAttack uses vulnerabilities of MEMS microphones. On the other hand, the Audio Hotspot Attack uses a physical phenomenon i.e.,non-linearity in the air. Audio Hotspot Attack countermeasures are therefore more difficult to create given they do not rely on any existing vulnerabilities.

2.9 Conclusion

In this work, we proposed a new inaudible voice command attack named "Audio Hotspot Attack." Its feasibility was evaluated through extensive user studies and reproducible experiments. We demonstrated that when directional sounds are emitted from parametric loudspeakers and not perceived by a nearby person, attacks can succeed over relatively long distances (2–4 m in a small room and up to 10+ m in a hallway); further, these attacks are tolerant against environmental noises. Although the Audio Hotspot Attack is currently a proof-of-concept, possible countermeasures to render the threats unsuccessful have been provided. The proposed attack uses ultrasound self-demodulation, which is a parametric phenomenon. We believe that this concept sheds new light onto ongoing security research focused on mobile and IoT devices, from the viewpoint of acoustic inputs.

2.10 Additional Results of Section 5.1.2

We tested the Audio Hotspot Attack to several other devices, in addition to Google Home and Amazon Echo. We measured the maximum distance at which the attack succeeded in a hallway. All the experimental conditions are the same as those stated in Section 5.1.2. Table 2.4 summarizes the results. We note that as the smartwatch did not recognize the activation command ("Hey Siri") even when uttered by the owner, we skipped the activation command by manually launching the recognition mode and tested the attack. We notice that while the attack succeeded for all the devices, the maximum distance becomes smaller, as compared to the smart speaker. As these devices (smartphones and a smart watch) are handheld devices with small microphones, it is natural that the distance at which a speech can be recognized decreases even when a normal audible sound is inputted.

2.11 Additional Images of Section 5.1.2 and Section 6

Table 2.4 Longest distance at which the attack was effective in a hallway. Google assistant was installed to ASUS Zenfone, SONY Xperia and SHARP AQUOS SHV37 from google play. Siri was used in the experiment of Apple watch.

	Longest distance [m]	
Devices	Activation	Recognition
ASUS Zenfone 2	6.1	5.5
SONY Xperia Z4	1.0	3.9
SHARP AQUOS SHV37	1.1	2.2
Apple Watch	–	7.5

Fig. 2.18 Experiments in a hallway (left) and outside (right). We install the parametric loudspeaker and smart speaker at the same height. The detail result was presented in Sec 5.1.2, Table 2.

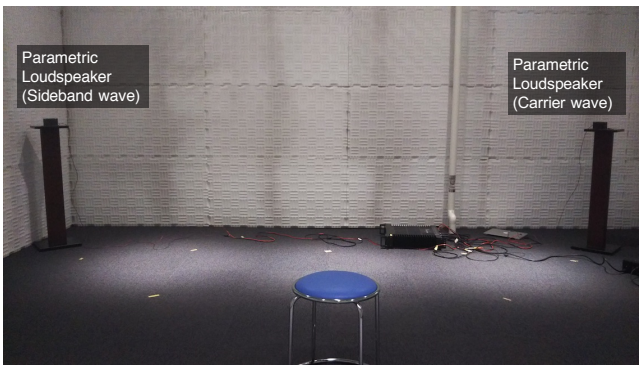


Fig. 2.19 A setup of the user study (cross attack). The left side of parametric loudspeaker emits the sideband wave, and the right side of parametric loudspeaker emits the carrier wave.

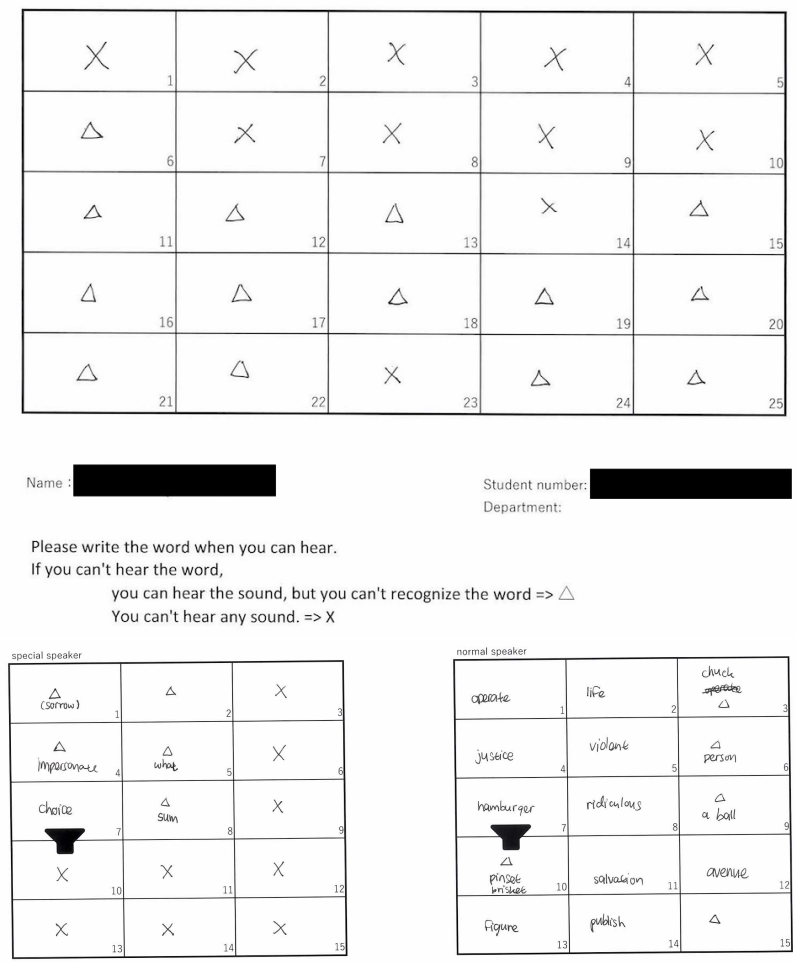


Fig. 2.20 Answer sheet reported by one of the participants. Top: answer sheet for cross attack. Bottom left: answer sheet for linear attack. Bottom right: voice attack with a dynamic speaker. The participants reported the word when they can hear. X means that "the participant cannot hear any sound", and △ means that "the participant hears the sound, but he/she cannot recognize the word."

Chapter 3

Contribution2: Preventing against threats caused by analog signals

3.1 Introduction

Cyber-physical systems (CPSs) are used to realize seamless interactions between physical and cyber spaces. Smart homes, robotic vehicles (e.g., self-driving cars and autonomous drones), medical devices, and various other IoT devices equipped with voice recognition capabilities are promising examples. Although the widespread adoption of CPSs can result in substantial benefits to society, many security threats that exploit the physical sensors inherent in CPSs [2, 3, 4, 5, 6, 7, 8, 9, 10] have been identified.

The threats range from bringing unauthorized input to the sensors of the CPS, causing it to make malicious behaviors, to taking control of the device itself; e.g., the threat of analog signals generated by commands from IFTTT [67] applications, becoming inputs to other CPS devices and causing malicious behavior [20], and the threat of injecting inaudible voice command into voice assistant devices by modulating voice signals into the laser output [6], or capacitors [10].

Thus, as application-centric CPS services such as IFTTT and Voice App become more prevalent, the risk of unauthorized control of CPS devices increases, making it difficult for device developers and users to be aware of such threats.

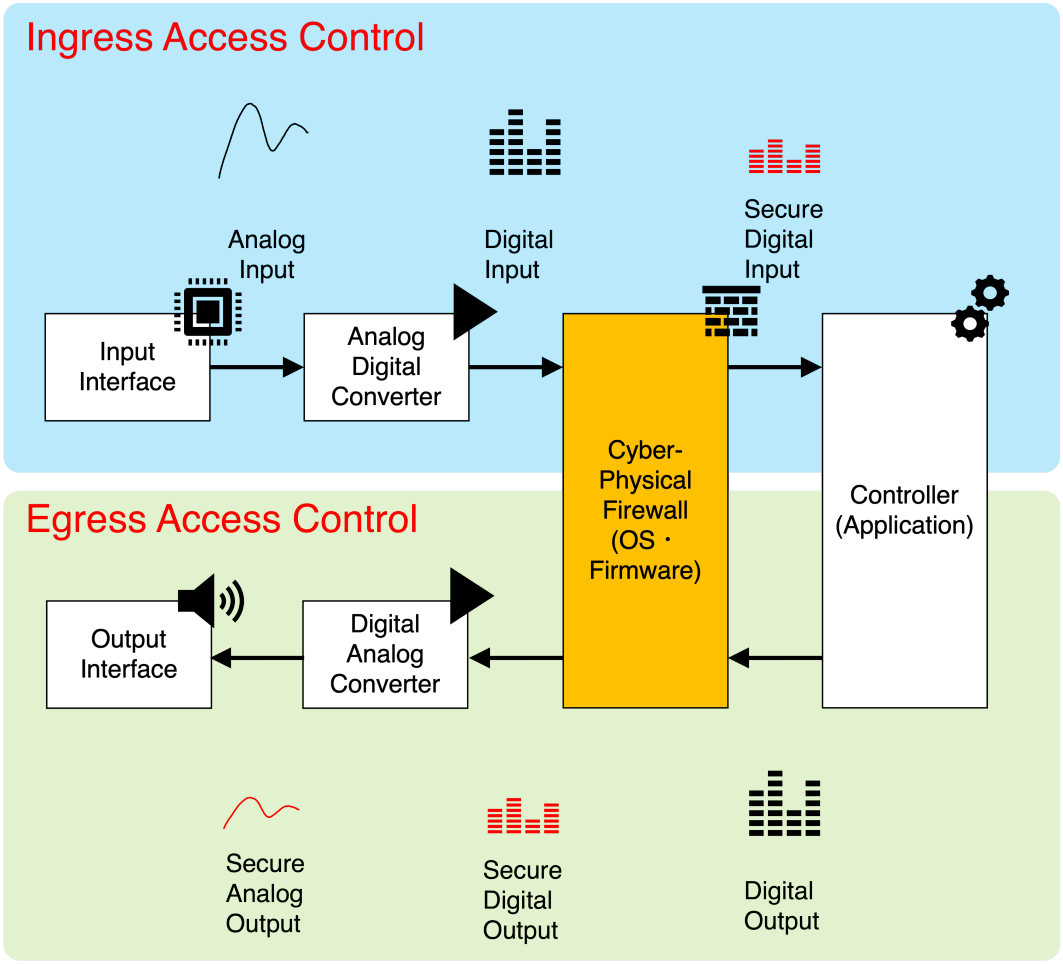


Fig. 3.1 Overview of the CPFW framework for the ingress access control of input signals (top, previous studies) and egress access control of output signals (bottom, our research).

To mitigate these threats, previous studies typically adopted an approach of detecting and mitigating attacks by analyzing sensor input [2, 21, 22, 4, 23, 8, 24, 25, 26, 27]. These countermeasures usually focused on sensor input data regardless of sensor type, presenting the following shortcomings: (1) Attack detection accuracy may be reduced considerably by the noise generated in the physical space [31, 4, 13] and (2) It is difficult to accurately detect an attack that exploits circuit nonlinearity of input-processing data [32, 8, 33] (3) Input-based approaches do not directly block

Table 3.1 Features of Ingress and Egress Access Control.

Features	Ingress	Egress
Effect of difference of noise or environment	△ Mitigated by noise filtering	○ No output
Effect of A/D converter	× Nonlinearity exists	○ Not via A/D converter
Versatility of countermeasures	△	○ Solved in this paper
Realtime processing	△	○ Solved in this paper
Scalability and Sharing knowledge	○ Numerous previous researchs	○ Solved in this paper
Negative effects of analog signals on humans (hearing loss, burns, etc.)	× Cannot prevent	○ No output

the source of the attack.

In order to address the above issues, we developed a framework named "Cyber-Physical Firewall" (CPFW) to provide an access control mechanism for analog signals. The unique feature of CPFW is that it adopts an approach to implement "egress access control," which aims to employ the policy enforcement on the attacker's or exploited device so that it will not emit a malicious analog signal to the physical space[*1] (Figure 3.1). Specifically, it detects and regulates signals with patterns not originally intended to be emitted from the device, and are therefore likely to be exploited by an attacker, before they are actually output into the physical space, with the goal of defeating the security threats described above.

CPFW targets the digital signal before output as an analog signal. This approach avoids the factors that make threat detection difficult, i.e., attacks based on nonlinearities inherent in the device receiving the signal and the effects of environmental noise. CPFW also has the advantage of embedding a defense mechanism directly into the device, which can be exploited by an attacker and potentially become the source of an attack, thereby realizing a root cause countermeasure. The advantages and disadvantages of ingress and egress access control are shown in Table 3.1. Egress access control enables control that is difficult to address by ingress access control.

Typical access control mechanisms, such as firewalls and secure operating sys-

[*1] Like the conventional approaches, the CPFW framework also supports "ingress access control, " which apply the policy enforcement to the analog signal received by a victim's device.

tems, generally target structured digital data. The challenge of CPFW is to realize a general and flexible access control mechanism for unstructured analog data with a high degree of freedom, which is sent and received in physical space. To tackle such a challenge, we first define the requirements and design specifications necessary to realize the CPFW framework. The key ideas are the partitioning of input data into blocks, feature extraction for analog signals, adoption of policy-based access control, and graphical policy description interface. We implement a prototype of the CPFW framework based on the defined specifications, and demonstrate that access control for analog signals can be properly realized and that real-time processing is possible. We will demonstrate that the CPFW framework can prevent real-world attacks against CPS devices, such as ultrasonic wave attacks (DolphinAttack [8]), noise attacks (Audio Adversarial Example [3]), and resonant attacks (WALNUT [7]).

The contributions of this study are summarized as follows:

- We developed the CPFW framework to provide generic and flexible access control for regulating malicious analog signals targeting CPS devices.
- The egress access control approach facilitates overcoming several challenges, e.g., noisy environments, exploitation of nonlinearity of data processing circuits, and direct regulation of the attack sources, which are otherwise difficult to address with conventional ingress access control approaches.
- Extensive experiments with a prototype implementation were employed to demonstrate the feasibility of the CPFW framework.

3.2 Threat Model

In this section, we first present our target, a CPS device and its inherent security threats. An assumption regarding the attacker's resources and abilities is then made.

3.2.1 Target CPS device

Cyber Physical System (CPS) makes decisions and performs actions based on the data exchanged between Physical Space and Cyber Space. CPS devices include IoT

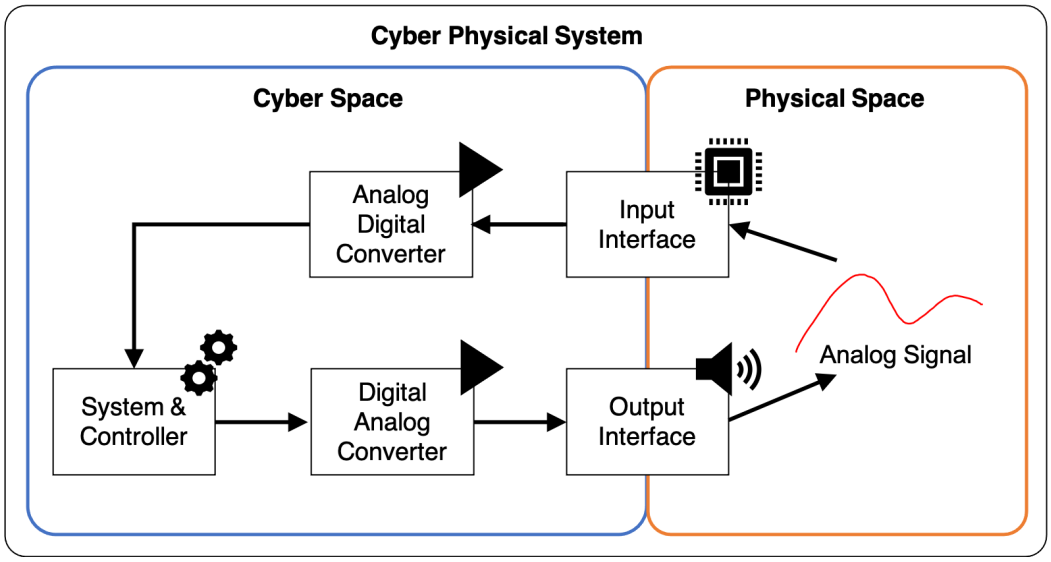


Fig. 3.2 Overview of a CPS device.

home appliances like voice assistant systems, medical devices like health monitoring sysmtes, and driver assistance and self-driving cars.

As shown in Figure 3.2, a CPS device has three key elements: input interfaces (e.g., sensors, microphone, and camera), controllers, and output interfaces (e.g., speaker, light display, and motors). Input interfaces read analog signals generated in the physical space and send them to the controller. Output interfaces receive analog signals sent from a digital-to-analog converter and output them to the physical space. A controller receives data sent from the input interfaces, process it, and generates the output through the output interface. Notably, security threats and attacks conducted between cloud services and devices are outside the scope of this research.

3.2.2 Target Security Threats

In this study, we focused on the security threats caused by the malicious analog signals, which would be read by the input interfaces of a CPS device, leading to unintended/non-authorized behaviors. Stealth voice command injection attacks such as DolphinAttack [8], Light Commander [6], audio adversarial examples (AEs) [3] , are typical examples of such attack. These attacks aim to inject malicious voice

Table 3.2 Classification of Security Threats.

threat types	threat name
Ultrasonic	DolphinAttack [8], Inaudible voice commands [33], Audio Hotspot Attack [68]
Noise	Audio Adversarial Examples [3], Hidden Voice Commands [69], Jamming [11]
Resonant	WALNUT [7], Rocking Drones [5]

commands unbeknownst to the device owner. Another example is the misuse of sonic waves to inject malicious noise into the sensors of autonomous vehicles. For instance, Son et al. demonstrated that drones could be attacked with sound waves in a frequency range that can be emitted by off-the-shelf loudspeakers [5]. Table 3.2 shows the classification of the attack methods using analog signals, including the research cases mentioned above. The three types of attacks shown in Table 3.2, i.e., ultrasonic, noise, and resonance, are typical covering a wide range of attacks using analog signals.

The goal of this study is to develop a firewall that can mitigate the malicious analog signals, such as the ones described above. As mentioned in the introduction, the originality of our approach lies in the fact that it provides egress access control. In other words, even if an attacker attempts to exploit an off-the-shelf device to emit the malicious signal, the built-in CPFW mechanism can prevent the attack. In such a scenario, the device manufacturer can pre-configure the CPFW according to the device's intended use and possible misuse patterns; For example, a loudspeaker manufacturer may want to inhibit the generation of very high frequency sound waves that would be inaudible to humans but could be used for inaudible voice command attacks. Moreover, CPFW can be applied to the input signal as well, that is, CPFW can be configured on the victim device.

3.2.3 Attacker's Resources and Ability

In concurrence with the majority of recent studies on the security threats of CPS devices, we believe that it is a reasonable assumption that an attacker would use/-exploit off-the-shelf equipment to emit malicious analog signals. In fact, previous studies on audio AEs [3], replay attack [4], rocking drones [5], and WALNUT [7]

were demonstrated using off-the-shelf equipment, which were all inexpensive and commercially available.

We assume that when an attacker wants to use external hardware to emit malicious signals, they cannot directly embed an adversarial circuit into a target device circuit, but they can connect an adversarial circuit that generates malicious signals to the input interface of the target device. We also assume that the CPFW is implemented such that it is tamper-resistant and cannot be bypassed to emit signals. An attacker can also install a malicious application on the equipment to emit malicious analog signals; i.e., the attacker may use software for performing the attack. Cases wherein an attacker leverages custom-developed hardware, which does not allow developers to pre-install our CPFW framework, are outside the scope of this research.

We consider two main attacker objectives.

1. Spoofing: An attacker inserts incorrect values to sensors. (e.g. replay attacks: recording another person's voice to use for authentication)
2. DoS: An attacker interferes with sensor readings. (e.g. resonant attacks: using ultrasonic waves to interfere with accelerometer values)

In the CPFW framework, we provide a suitable countermeasure regardless of the purpose of attack.

3.3 Design of Cyber-Physical Firewall

In this section, we first clarify the requirements of the CPFW framework (i.e., what to develop). Then, we present the design specifications derived from the requirements (i.e., how to develop). Finally, we outline the overall architecture to be built based on the design specifications.

3.3.1 Requirements Specification (What)

To meet the objectives of the CPFW framework outlined in Section 3.1, we set the following three fundamental requirements, **R1**–**R3**.

R1: Real-Time Data Processing

The CPFW framework inspects the received data, and if it finds that a policy is violated, it applies the appropriate control, such as noise reduction or lowpass filter, and transmits the data to the subsequent stages. Therefore, to avoid reducing the quality of service, it is necessary to operate in real time.

R2: Flexible Access Control

As mentioned earlier, the physical space has a high degree of freedom, and so does the freedom of attacks using analog signals. Therefore, the access control mechanism must be flexible and generic so that it can cope with various attack patterns. Similarly, because there are a variety of devices to which the CPFW framework can be applied, it is necessary to be flexible enough to support the various input–output (I/O) interfaces installed in the devices.

R3: Extensibility

Ideally, the necessary policies will be built into the device beforehand based on the principle of security-by-design. However, in reality, it is common for new threats to arise after the device has been shipped. Therefore, it is essential that the CPFW framework be extensible so that it can cope with new threats that may arise in the future.

3.3.2 Design Specification (How)

Next, based on the three requirements shown above, we derived the following design specification (**D1**–**D4**):

D1: Block-based data processing

In response to requirement **R1**, we adopted an approach that divides the input data stream to the CPFW into blocks and processes the data on a block-by-block basis. Here, a block consists of N sample points (referred to as *frames*). N is a parameter that is determined by the trade-off between the number of samples required to statistically determine whether an analog signal violates a policy and its tractable size for real-time processing.

D2: Attribute definition/extraction

In response to requirements **R1**, **R2**, and **R3**, we developed a method of defining and extracting the information (attributes) needed to apply policy-based control

from the blocks. We can define various attributes based on statistics obtained from time series data, such as mean and variance, statistics obtained from frequency analysis, and outputs obtained by applying advanced data processing mechanisms, such as speech recognition to the signals contained in blocks. Notably, the attribute extraction process must be fast so that it can work in real time.

D3: Policy-based access control

In response to requirement **R2**, we adopted a policy-based access-control approach. This approach is a common method that has various applications, such as network firewalls and secure operating systems. The advantage of this approach is that arbitrary policies can be defined by using a data process description model, which we introduce as the specification **D4**. We adopted the *if–then rule* as a method of realizing policy-based access control and implemented a policy violation detection (*if*) and a policy enforcement mechanism (*then*).

D4: Policy description interface

In response to requirements **R2** and **R3**, we introduced a policy description interface for expressing policies. In particular, we adopted a graphical user interface (GUI), such as the GNU Radio Companion GUI [70]. The adoption of a GUI has the advantage of being intuitively configured for complex data processing, consisting of multiple signal processing. The description interface has the advantage of enabling effective policy sharing and reuse.

3.3.3 Overall architecture with integrated design specifications

Figure 3.3 shows the overall architecture of the CPFW framework, which integrates the aforedescribed design specifications. The CPFW first receives the data and divides them into blocks consisting of N frames (D1). Next, it extracts attributes for each block (D2). The attributes to be retrieved are defined in a policy written using a policy description interface (D4). The framework checks whether the extracted attribute violates the policy and executes policy enforcement if it finds a violation (D3). The process is completed by outputting the final block as a data stream (D1).

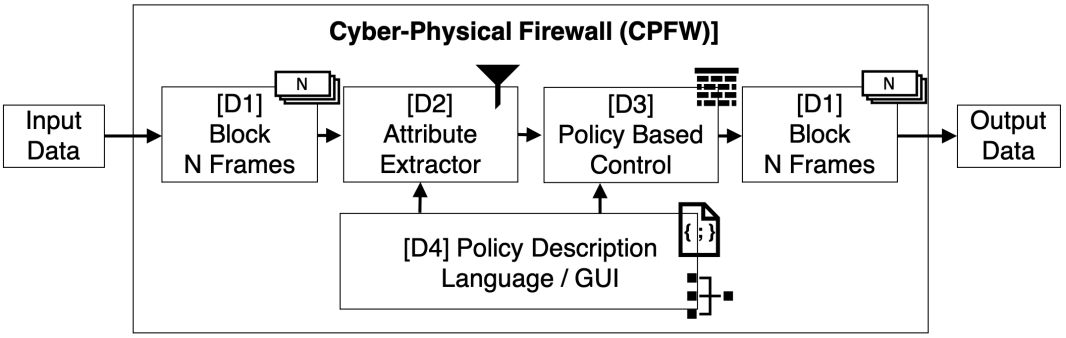


Fig. 3.3 Overall architecture.

Table 3.3 List of General attribute and their usage.

Class	Attribute name	Usage
Base Info	Amplitude	Volume
	FFT	Frequency distribution analysis
	Sampling Rate	Determine the max output frequency
Power	Total Power	Total power
	Mean Power	Mean Power
Frequency	Mean frequency [71]	Mean of frequency distribution
	Median Frequency [71]	Median of frequency distribution
	Peak Frequency [71]	Derive the strongest frequency
	Variance of Central Frequency [71]	Variance of frequency distribution
	Max Frequency	Detect maximum frequency
Rate	Threat Frequency Rate	Derive percentage of threat signal power
	Zero Crossing Rate [72]	Voice Part Extraction
Energy	Short Time Energy (STE) [73]	Voice Part Extraction

3.4 Descriptions of System Implementation

We present the implementation details of the CPFW framework according to the design specifications shown in section 3.3. The following description assumes the egress access control, that is, we adopted a case wherein the CPFW framework

Table 3.4 List of attacks that can be countered by the policy-enforcement schemes.

Enforcement scheme	Countable attacks
Decrease Amplitude	Replay Attack [4], Voice Synthesis [4], WALNUT [7], Rocking Drones [5]
Lowpass Filter	DolphinAttack [8], WALNUT [7], Rocking Drones [5], Audio Hotspot Attack [68]
Noise Reduction	Audio Adversarial Examples [3], Hidden Voice Commands [69], Jamming [11]

aims to prevent the emission of malicious analog signals from the device owned or exploited by an attacker.

3.4.1 Implementation of block-based data processing (D1)

To achieve block-based data processing, we sample and process the digital input data at a sampling frequency of f, which can be set to any value according to the signal processing capability of the device. In this study, we used values of $f = 48$ or 96 kHz. The number of quantization bits is fixed at $q = 16$ bits. The number of frames, which comprise a block is set to a power of two, which is suitable for applying a fast-Fourier transform (FFT). In this study, we adopted $N = 1,024$ as the result of preliminary experiments using a Raspberry Pi 3 Model B+. We confirmed that $N = 2,048, 4,096$ is also suitable.

3.4.2 Implementation of attribute definition/extraction (D2)

Using a set of values recorded in N frames, various statistical values can be calculated as attributes. For example, common statistical values, such as mean, variance, median, maximum, and minimum, can be used as attributes. One can also apply FFT to obtain statistics based on frequency analysis, such as average frequency, median frequency, variance of central frequency, average power, peak frequency, and maximum frequency. Statistics specific to speech recognition, such as zero

crossing rate, can be calculated and used as attributes. Table 3.3 summarizes the list of general attributes that can be extracted from the observed signals and their usage. We have confirmed that all these attributes can be extracted with low latency through experiments using the Raspberry Pi.

A policy can be defined by using a combination of these statistical values. For example, to define a policy for regulating the emission of ultrasonic waves that could be exploited for hidden voice-command injection attacks, such as DolphinAttack [8], the average frequency can be used. If the value exceeds 20 kHz, we consider that the policy is violated.

When a signal having a specific frequency is subject to regulation, we can introduce several metrics, such as threat frequency rate (TFR), which is defined as the ratio of the threat signal to the total power for N frames. Assuming that the threat signal is detected in the range of $[f_L, f_H]$ Hz, and $X(n)$ is the n-th component of the FFT result, TFR is computed as

$$TFR(f_{DL}, f_{DH}) = \sum_{n=F^{-1}(f_{DL})}^{F^{-1}(f_{DH})} X(n) \Big/ \sum_{n=0}^{N} X(n), \tag{3.1}$$

where $F^{-1}(f)$ is a function that extracts the component number of the FFT result from a given frequency, f.

In addition to the statistical values calculated from the frames, the output of high-level data processing can be used as attributes. For example, if the target is a speech signal, we adopt the outputs of speech recognition (i.e., text data) as the attributes. For the recognition text, we used the results obtained from the Google speech-to-text API [74]. The text output can be used to detect voice command attacks, which make use of wake-up words such as "Ok Google" or "Alexa." The text can also be used to prevent the output of privacy-sensitive or offensive words by applying the NG word lists.

3.4.3 Implementation of policy-based access control (D3)

The policy-based access control is implemented according to the *if–then rule*, which comprises policy violation detection (*if*) and enforcement (*then*). During the detec-

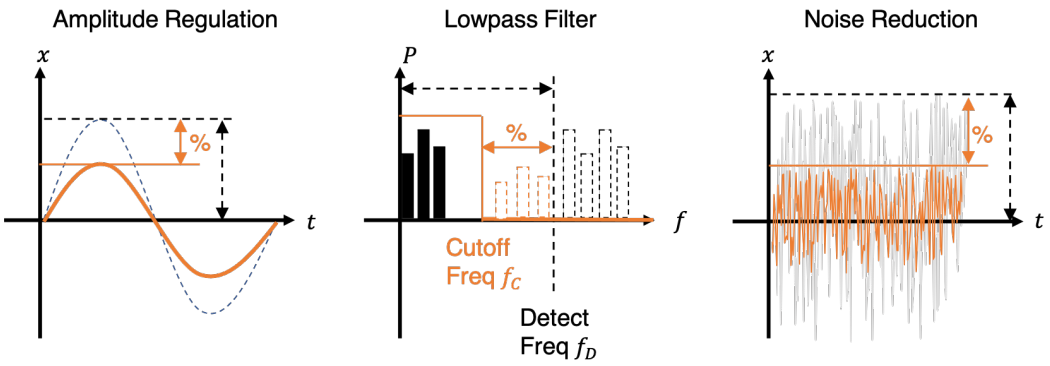


Fig. 3.4 Overview of enforcement schemes. '%' represents the enforcement level.

tion phase, we use criteria that can be preset or adjusted per user. In Section 3.5, we present an example case to set the criteria (threshold) for detecting ultrasonic attacks.

Although various methods can be employed for policy enforcement, this paper describes three typical methods: signal-strength attenuation, lowpass filters, and noise removal [75]. An overview of each enforcement method is shown in Figure 3.4.

Enforcement level is defined as a real number between 0 and 1, where 0 represents no policy enforcement, and 1 represents the maximum policy enforcement. We can adjust the level to meet the condition that the enforcement is effective and does not damage the quality of service. When amplitude regulation is used, the enforcement level is defined by the ratio of the amplitude after regulation to the original amplitude. In the case of lowpass filter regulation, the enforcement level is calculated as $(f_D - f_C)/f_D$, where f_D is the frequency at which regulation is applied, and f_C is the cutoff frequency at which regulation is required. In the case of noise removal, the enforcement level is defined by how much the amplitude of the noise to be removed is attenuated relative to the noise amplitude. When level = 1, the noise is completely removed. When level = 0.5, the noise signal is processed so that the amplitude of the noise is reduced to half.

We summarize the attacks that can be countered by the enforcement schemes in Table 3.4.

3.4.4 Implementation of policy description interface (D4)

MATLAB's Simulink [76] and the GNU Radio Companion [70] are well-known systems that allow developers to define complex data-processing flows using an intuitive GUI. This study is inspired by these existing systems and implemented a graphical analog policy diagram (APD) description interface. The main components of the APD are shown in Figure 3.5 (top). An example implementation of a policy using the mean frequency as an attribute is shown in Figure 3.5 (bottom). APD defines a description language internally, which can then be linked to any external programming languages.

We defined Analog Policy Language (APL) as an intermediate language for converting APD into processable if-then rules program. Listing 3.1 shows the definition of APL in Backus-Naur Form (BNF).

Figure 3.6 demonstrates the relationship between APD, APL, and the policy system. Policy description is a sentence written in natural language. The policy description preset for sound is shown in Table 3.5. These descriptions summarize the elements that are assumed by the sound security and privacy threats presented in previous studies. To implement the policy description, users select the attribute, threshold, and enforcement method, and convert them to if-then rules. If-then style policies convert to APD by using if-then block in Figure 3.5. We show the APD policy example in Figure 3.7.

The block diagram described in APD is converted to JSON format APL and loaded into the policy system. The framework can be used in the following three ways by using APD and APL.

1. Use preset policies (Examples in Table 3.5)
2. Adapt the parameters of the preset policy to user's environment.
3. Create a new policy (customized policy)

Figure 3.7 presents an example of APL and APD, wherein LPF is employed to regulate signals if the mean frequency exceeds 20 kHz.

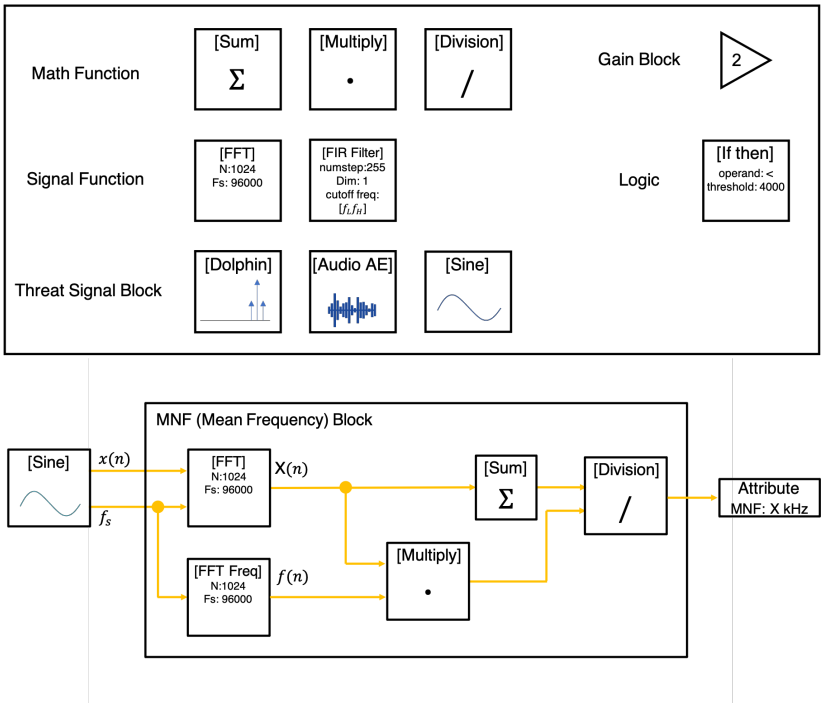


Fig. 3.5 APD basic design (top) and example of the APD structure of mean frequency block (bottom). f_s is the sampling frequency, and $f(n)$ is the frequency of frame n.

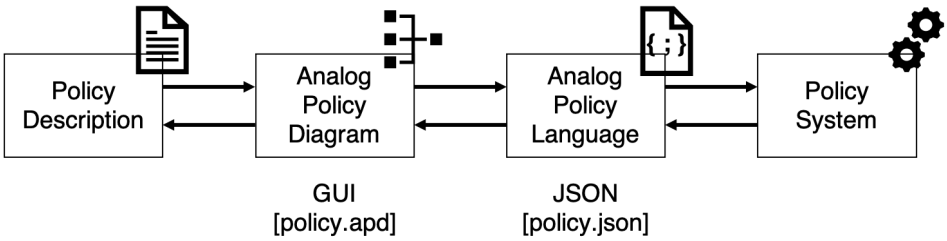
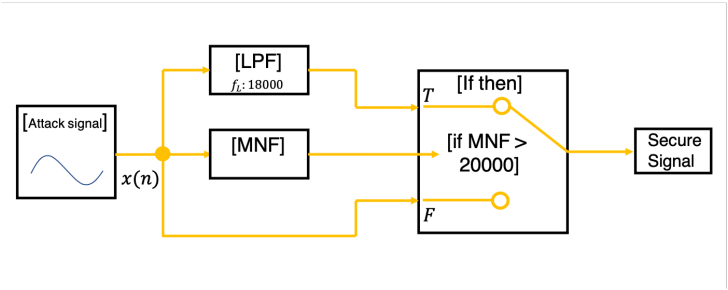


Fig. 3.6 Overview of the implementation of policy description Interface.

Listing 3.1 Analog Policy Language (APL) in BNF. “+” indicates one or more occurrences and “*” indicates zero or more occurrences.




```
{
 "attributeExtractor": {"function": "getMNF", "input": "NFrames"},
 "policyDetector": {"attribute": "MNF", "operator": ">=", "threshold": 20000},
 "policyEnforcement": {"enforcement-method": "BandpassFileter",
                       "f_L": 18000, "f_H": "inf",
                       "enforcement-level":0.1}
}
```

Fig. 3.7 Examples of APD (top) and APL (bottom), in which an LPF is used to regulate signals if the mean frequency exceeds 20 kHz. The top figure shows an example of passing a signal through LPF when the if-then block condition is true (T) and passing raw output when it is false (F). APD (top) is converted to APL format (bottom).

⟨*policy*⟩ ::= ⟨*attributeExtractor*⟩⟨*policyDetector*⟩⟨*Enforcement*⟩

⟨*attributeExtractor*⟩ ::= [⟨*function*⟩⟨*input*⟩]+

⟨*input*⟩ ::= ⟨*attribute*⟩ | ⟨*Nframes*⟩

⟨*policyDetector*⟩ ::= [⟨*expression*⟩]+

⟨*expression*⟩ ::= ⟨*attribute*⟩⟨*operator*⟩⟨*threshold*⟩

⟨*operator*⟩ ::= >| <| ≤ | ≥ | in | =

⟨*Enforcement*⟩ ::= ⟨*function*⟩⟨*enforcement-level*⟩[⟨*argument*⟩]*

Table 3.5 Examples of egress access control policy description for audio signals.

Category	Class	Description	Condition examples for detection
Voice	Ultrasound [8, 33]	Restrict output sound frames that contain ultrasonic waves above 20 kHz.	getMNF(Nframes) ≥ 20000
	Noise [69, 11]	Restrict sound output with amplitude less than 0.1.	getAmplitude(Nframes) ≥ 0.1
	Replay [24, 13, 2]	Restrict output sounds that contain voice command.	getZCR(Nframes) ≥ 0.2
Privacy [77]	Information	Restrict output sound frames that contain voices.	getZCR(Nframes) ≥ 0.2
		Restrict voice output content to be delivered as message cards.	getZCR(Nframes) ≥ 0.2
Sensor	Resonant Attack [7, 5]	Restrict sound sources that exceed the specified frequency (ex. 4 kHz).	getMDF(Nframes) ≥ 4000

3.4.5 Specification of the prototype implementation

We implemented a prototype of the CPFW framework in Python running on Raspberry OS Lite. The hardware used is a Raspberry Pi 3 Model B+ (Element14, CPU: Cortex-A53 64-bit, 1.4GHz, memory: 1GB). We implemented a prototype of the CPFW framework in Python running on Raspberry OS Lite [78]. The prototype was implemented using a microphone with the Voice AIY kit [79] as the CPS input interface, a Raspberry Pi 3B+ (Element14, CPU: Cortex-A53 64-b, 1.4 GHz, memory: 1 GB) [80] as the controller, and a dynamic speaker with a 6 cm diameter as the output interface. We used `numpy` and `scipy` as our scientific computing libraries. The FFT algorithm used for the frequency analysis was SciPy's `rfft()` function. For noise reduction, we used the algorithms described in [75]. For audio I/O processing, we used the asynchronous processing mode of `pyaudio`. The total number of lines of code implemented was 492. The entire line is expected to increase depending on the number of preset attributes and enforcement methods. A photo of a prototype of the CPFW framework implemented on a custom CPS device is shown in Figure 3.8.

3.4.6 System Implementation Model of CPFW

We present a system implementation model of the CPFW framework. CPFW is implemented as a function of the OS or Firmware. Figure 3.9 shows an example of the system implementation model. In this model, the developer installs the CPFW in the OS or Firmware before the product is shipped. The functions and initial settings of the CPFW cannot be circumnavigated or modified by malicious

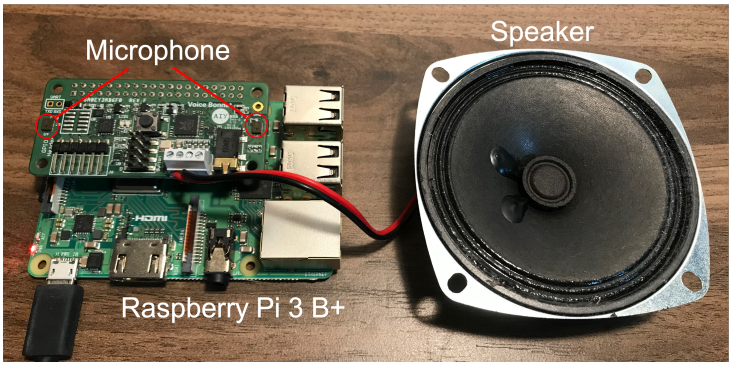


Fig. 3.8 An photo of the prototype implementation of the CPFW framework.

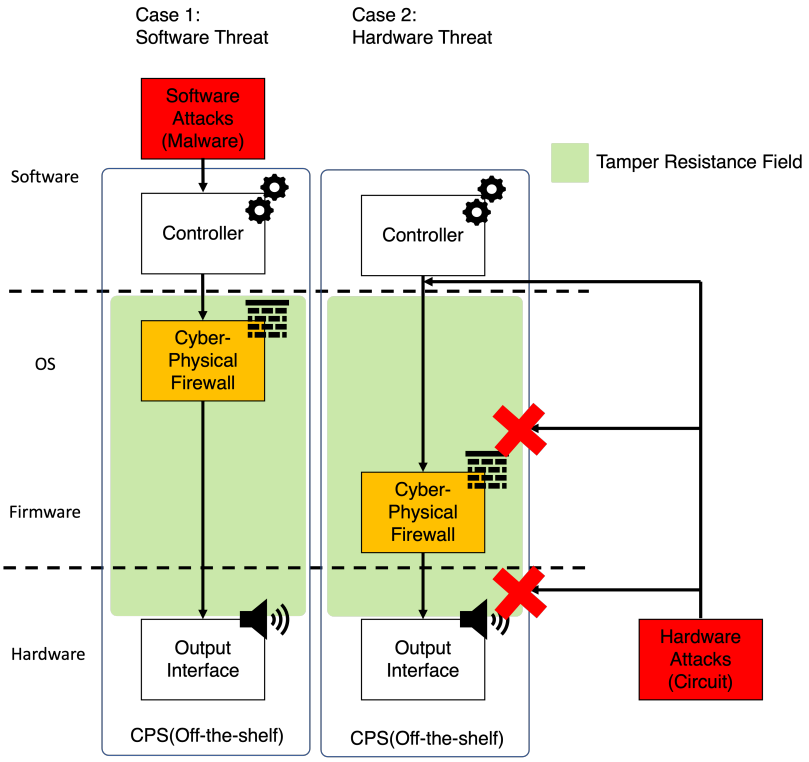


Fig. 3.9 Examples of system implementation of CPFW (OS and firmware) and two threat cases (software threat and hardware threat).

users. As a mechanism to prevent bypassing the CPFW functions, firmware TPM (Trusted Platform Modules) can be used to ensure that the input/output and internal processing of the CPFW is tamper-resistant, i.e., impossible to analyze.

In addition, the configuration of the CPFW can be changed by following the regular procedure shown below. Specifically, CPFW updates and settings must be verified by signature. Updates and settings with signatures are limited to only those license holders who can properly handle analog signals. The license for analog signals is the same as that granted for radio signals, and it is the right of a government agency or specific government authorized organization to grant permission to change the configuration of a CPFW to a specific person or entity that meets the required conditions.

Previously, utilization of the licensing system for the output of analog signals, except for radio signals, has not been considered. While licensing systems for regulation radio signal emission have been widely deployed in the world, similar licensing system for regulating the emission of generic analog signals has not been adopted. However, the threats to CPS devices caused by malicious analog signals have become vital these days. Therefore, device management under a licensing system appears to be a promising approach. The licensing and legal system for analog signals is discussed in further detail in Section 3.7.

Figure 3.9 shows an example of CPFW implementation and cases where CPFW is applied to a threat. Case 1 shows an example of a malware application running on CPS that creates a malicious analog signal threat. We prevent malicious signals from being emitted from the output interface by adding detection and regulation processing to device driver processing. CPFW can also handle case 1 with firmware implementation. Case 2 shows an attacker who connects an external circuit to the device and adds a malicious signal. In this case, it may not be straightforward to detect at the OS level. Implementing CPFW as firmware for CPS devices ensures that monitoring the added signal is possible at the lower level.

3.5 Feasibility Experiments

In this section, we detail the experiments performed to evaluate the feasibility of the CPFW framework. The evaluation was conducted from the following three perspectives: real-time performance, attribute extractor metrics testing, and policy violation detection accuracy.

3.5.1 Evaluation of real-time performance

We first evaluated the runtime overhead of CPFW to verify that countermeasures can be taken in real time (Requirement 1). We measured the following three runtime overhead items: (1) process time required to extract each attribute, (2) process time required to perform each enforcement method, and (3) end-to-end process time required from the beginning of attribute extraction to the completion of policy enforcement. For (3), we adopted a policy where output sound frames should not contain more than 20-kHz ultrasonic waves. To enforce this policy, we applied a low-pass filter with a cutoff frequency of 20 kHz. For experiments (1) and (2), we adopted a chirp signal as input. The chirp signal changes frequency from 0 to 30 kHz over 1 min. For the experiment (3) the input was a randomly generated 20+ kHz ultrasonic waves that lasts for 1 min.

Table 3.6 and 3.7 show the mean time required to extract each attribute and to perform each enforcement method, respectively. Figure 3.10 shows the end-to-end processing time of CPFW framework. These results demonstrate that the runtime overhead is very small and stable. For the end-to-end process time, the mean and maximum times required were 5.42 and 26.34 ms, respectively. According to the International Telecommunication Union's (ITU) Telecommunication Standardization Sector (T) standard for speech transmissions delay [81], a delay in the range of 0–150 ms is acceptable to most users. In the case of hearing aids, usability can be ensured if the delay of the hearing aid output is within 0–10 ms [82]. Thus, we conclude that the runtime overhead of CPFW application is sufficiently small.

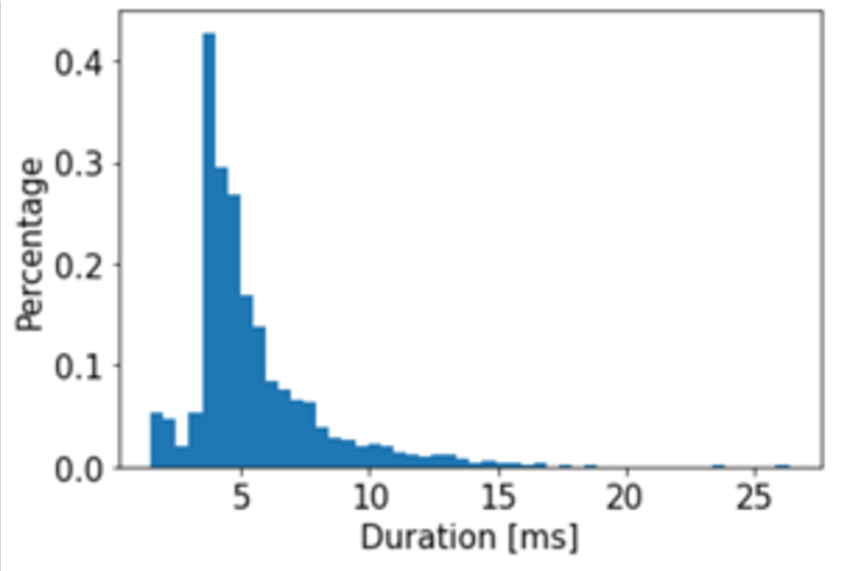


Fig. 3.10 End-to-end processing time measurement of the framework.

Table 3.6 Runtime overhead for attribute extraction per $N = 1024$ frames. Mean time m [ms] and standard deviation σ.

Attribute	m	σ
Amplitude	0.121	0.019
Sampling Rate	0.057	0.013
Zero crossing	0.260	0.048
FFT	1.258	0.240
MNF	1.385	0.271
MDF	2.677	0.451
TFR	2.192	0.452

Table 3.7 Runtime overhead for enforcement methods per $N = 1024$ frames. Mean time m [ms] and standard deviation σ.

Enforcement Method	m	σ
Decrease Amplitude	1.694	0.082
Lowpass Filter	3.219	0.140
Noise Reduction	8.471	0.689

3.5.2 Validity of the extracted attribute values

We verify that the attributes extracted by the CPFW framework could provide useful information for detecting attacks. The purpose of the evaluation was to verify whether the extracted attributes correctly capture an attack when a malicious audio signal is applied to a normal audio signal. As a normal audio signal, we used the first 30 s of J.S. Bach's Goldberg Variation Aria BWV988. Then, we prepared speech data in which a speaker utters "OK Google, What's on my next schedule?" The speech data is amplitude modulated with 25 kHz ultrasonic waves as the malicious signal. The length of the malicious signal was about 2.5 s. Notably, this malicious audio signal is a replication of the DolphinAttack [8] injected at 10 and 15 s after the start of the normal audio signal.

Figure 3.11 shows the results. First, in the top panel of the figure, we can observe that all three frequency statistical attributes (i.e., mean/median/max frequency) correctly captured the malicious audio signal around 25 kHz. Precisely, mean frequencies (MNFs) and median frequencies (MDFs) were more stable and responded precisely to the attack signal, whereas max frequency was more sensitive to noise and oscillated, even in areas where the attack signal was not present. Next, in the middle panel of the figure, we can observe that both threat-frequency rate (TFR) and zero crossing rate (ZCR) correctly captured the attack range. In summary, we confirmed the utility of the attributes by applying audio signals to the framework. Preliminary experiments demonstrated that it also works well with other analog signals.

3.5.3 Accuracy of policy-based access control

We evaluate the accuracy of policy violation detection using sine waves. The policy to be set is simple: reject sound waves above 4 kHz, which reflects the case study shown in Section 3.6.3. We adopt TFR, which was defined in Eq. 3.1, as an attribute to detect the violation. We experiment and compare results for two scenarios: (1) **Egress access control**: the CPFW framework runs on the attacker's or exploited

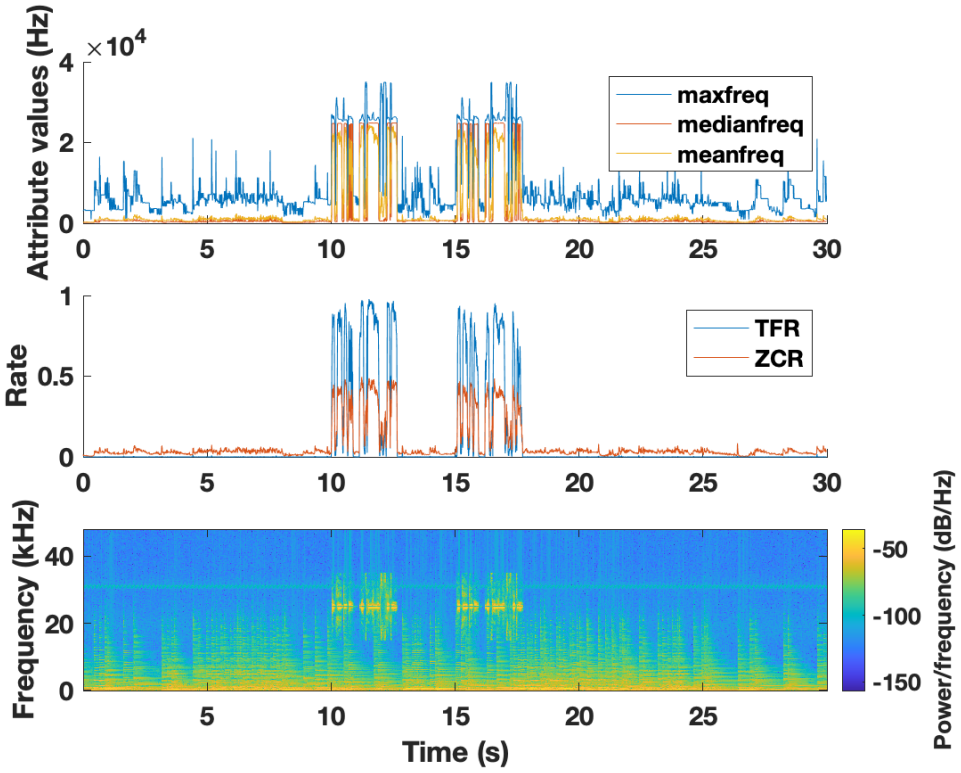


Fig. 3.11 Top: attributes based on frequency statistics, Middle: TFR and ZCR, Bottom: spectrogram of the original sound wave.

device and detects a policy violation *before* outputting the signal as an analog signal. (2) **Ingress access control**: the CPFW framework runs on the victim's CPS device and detects a policy violation *after* the output analog signal has been generated on the attacker's device and delivered/input to the target CPS device. To evaluate the accuracy, we adopt the ROC (Receiver Operating Characteristic) Curve [2]. We note that the ROC curve can also be used to extract useful information for adjusting the threshold.

We first generate sound wave data for the two scenarios as follows:

(1) **Egress access control:** We generated 3 seconds single-tone sine waves at frequencies from 1 Hz to 8 kHz at intervals of 1 Hz.

(2) **Ingress access control:** While the egress access control scenario does not involve the external noises, we need to consider the intrinsic noise that arise in the

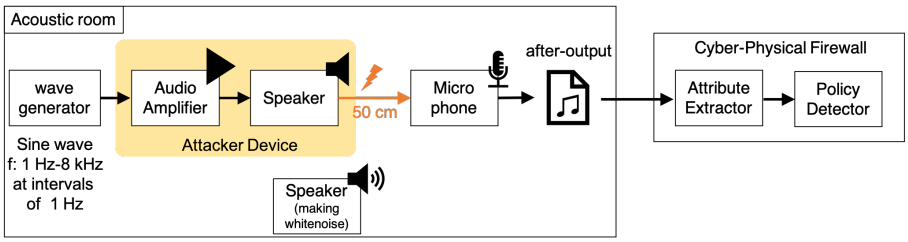


Fig. 3.12 A setup for generating the sound waves of the ingress access control scenario.

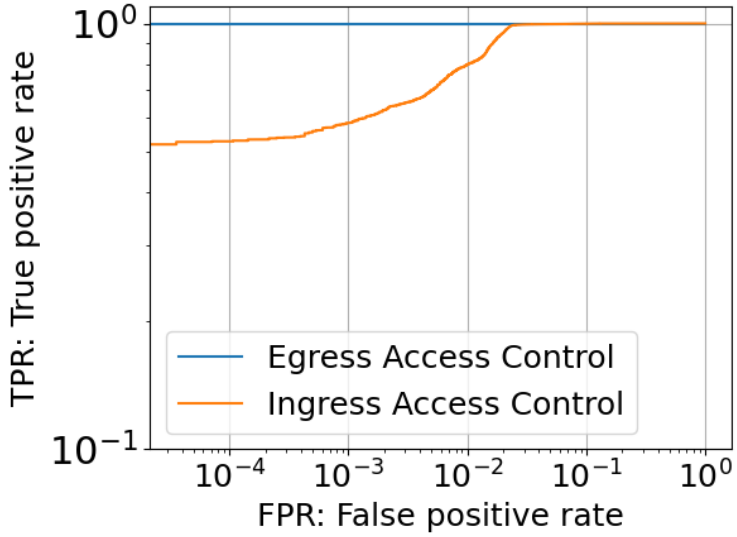


Fig. 3.13 ROC curves for the two scenarios: egress and ingress access controls.

physical space for the ingress access control scenario. Figure 3.12 presents a setup to generate the data for the ingress access control scenario. Each sine wave was played from a loudspeaker [50], and recorded at a distance of 50 cm in the dedicated acoustic experiment room. We changed the average noise level from 25 dB to 80 dB.

For each of the generated sound waves, we assign labels to the corresponding frames, i.e., 'T' for violation and 'F' for non-violation.

Figure 3.13 shows the results. In the case of egress access control, the policy violation detection was successful without error. This high success rate is attributed to the fact that no physical noise was applied inside the system; the high accuracy is a notable advantage of the egress access control approach. On the other hand,

the accuracy of ingress access control in detecting policy violations is degraded. Especially when the required FPR is low, the achievable TPR becomes low, implying that the framework tends to fail the policy violation detection. For example, when FPR = 0.01, TPR = 0.800, and when FPR = 0.001, TPR = 0.584.

3.6 Case Study

In this section, we demonstrate that the CPFW framework can mitigate real-world threats. Specifically, we targeted three types of attacks shown in Table 3.2, i.e., noise attacks (Audio AE [3]), ultrasonic attacks (DolphinAttack [8]), and resonant attacks (WALNUT [7]). For the audio AE, we employed noise reduction to eliminate the maliciously crafted perturbation. For the DolphinAttack, we employed bandpass-filter regulation so that no ultrasonic waves with maliciously modulated voice commands were emitted. Finally, for WALNUT, we employed a bandpass filter to regulate audio signals that attempted to perform a resonant attack on the accelerometer used in self-positioning systems. In the following experiments, we implement the CPFW framework as egress access control, i.e., we evaluate the effectiveness of the framework by employing the policy enforcement to the generated malicious signals before they are output from the attacker's or exploited device.

3.6.1 Preventing noise attacks

In this section, we took countermeasures against audio AE [3] as a typical example of noise attacks. We show the results of applying noise reduction to the audio AE. In this experiment, the generation of audio AE, policy enforcement by the CPFW framework, and speech recognition of the final output audio is performed in software. In other words, this is not an over-the-air experiment where audio is transmitted in physical space. The software experiment approach eliminates the effects of noise in the physical space, and allows us to simulate the ideal conditions for a successful attack. We show that the CPFW framework works well under such conditions.

As the audio AEs, we adopt the dataset developed by Carlini et al. [3]. The analysis of the dataset revealed that the average amplitude of the AE noise is 4.5×10^{-4} and

Table 3.8 Speech recognition results for each audio data.

Data	DeepSpeech	Google Speech to text
original	without the dataset the article is useless	without the data set the article is used
AE	ok google open evil dot com	without the data set the art of course Eustis
denoised AE	without the davaset nordclice waspes	without the data set the articles used to

has the equal intensity at different frequencies. Based on these results, we conclude that the adversarial perturbations have characteristics similar to white noise. We prepared a mask based on the enforcement level and the distribution of white noise. We then apply a mask to remove the white noises.

Figure 3.14 presents the spectorograms of the original audio (top), audio AE (middle), and the audio AE after the policy enforcement (bottom). Table 3.8 presents the results for speech recognition for each data. We used DeepSpeech [83] and Google Speech-to-Text [74] as the speech recognition implementations. For the DeepSpeech model, we used the one trained by Carlini [3]. Table 3.8 shows the results of the speech recognition. The audio AE generated by injecting perturbations to the original sound source data has succeeded in altering the speech recognition results. And by applying the policy enforcement to the audio AE, the speech recognition results are almost restored to the original recognition results. We found that the output of speech recognition by DeepSpeech is sensitive to the input data. Therefore, as a comparison, we examined the speech recognition results using Speech-to-Text on the same data. In all cases, the recognition results were close to each other. We deem that the reason for the lack of an exact match is due to the fact that the original audio is a bit unclear. To summarize, these results demonstrate that our framework successfully mitigated the threat of audio AE attack without compromising the original speech information.

3.6.2 Preventing ultrasonic attacks

The policy enforcement method for preventing the ultrasonic attacks was a low-pass filter that cuts off frequencies above 20 kHz. We reproduced the Dolphinattack [8]

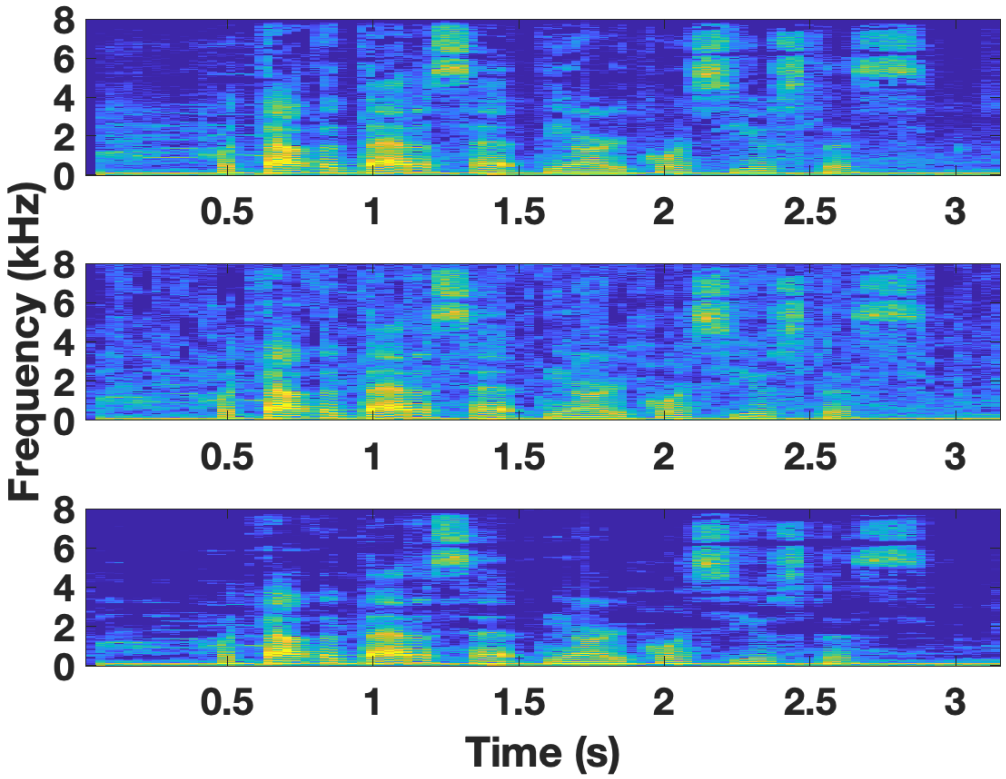


Fig. 3.14 Spectrograms of the original audio (top), audio AE (middle), and audio AE after the policy enforcement (bottom).

as an example of ultrasonic attacks, and verified the countermeasures. We evaluated the effectiveness of the countermeasure by the results of spectrogram and speech recognition results even for the over-the-air situation. The recording of DolphinAttack was performed with a distance of 50 cm between the microphone and the attack speaker.

The ultrasonic signal of DolphinAttack [8] is generated by the amplitude modulation of the voice data (transcript: "OK Google, what's on my next schedule") at the frequency of 40 kHz, which causes inaudible voice command injection by exploiting the nonlinearity of the microphone circuit. Figure 3.15 presents the observed spectrograms. We see that the waveform caused by the DolphinAttack is no longer observable after the policy enforcement. In fact, the speech-to-text service [74] did

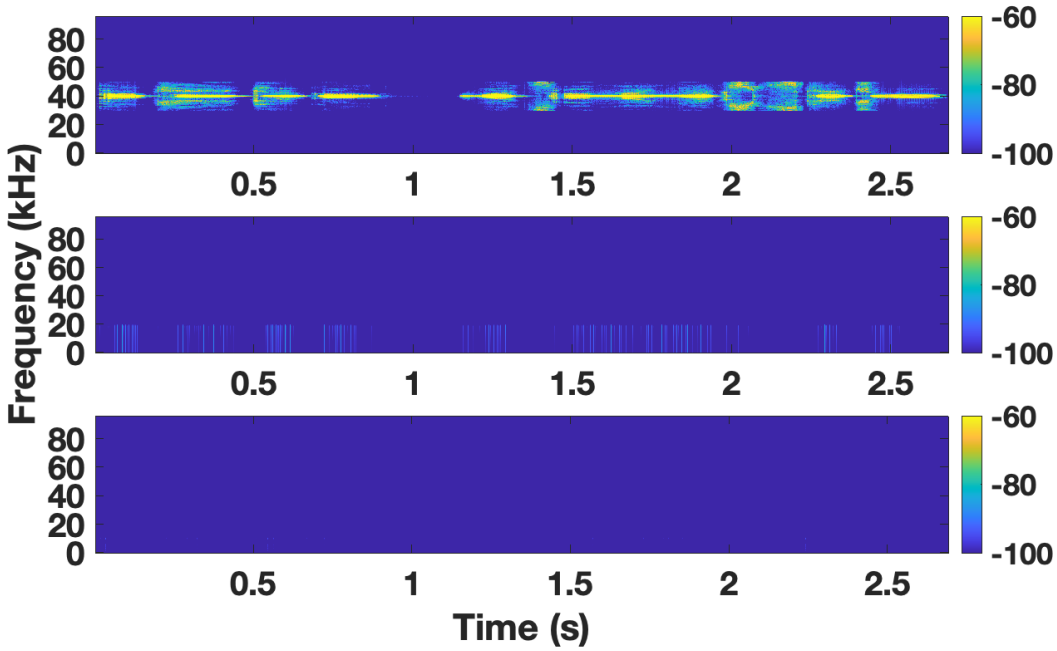


Fig. 3.15 Spectrograms of the DolphinAttack signal (top), policy-enforced signal before transmission (middle), and denoised audio signal after over-the-air transmission (bottom).

not detect the recorded sound as a voice. Thus, the CPFW successfully prevented the emission of DolphinAttack.

3.6.3 Preventing sensor resonance attacks

In this experiment, we prevent a sensor resonance attack on an accelerometer [7]. The experimental setup is shown in Figure 3.16.

First, to reproduce the resonance attack performed by the WALNUT [7] attack, we adopted a 9-axis sensor module, MPU9250 [84], which is equipped with an accelerometer, a gyro sensor, and a magnetometer. The speaker [85] and the amplifier [86] was used to emit the sine wave to cause resonance and was set 10-cm above the top of the target sensor. The loudspeaker emitted a pure-single tone between 50 Hz and 30 kHz at intervals of 50 Hz, and the sensor value at the time of each emission was recorded. As shown in the figure, the acceleration values were

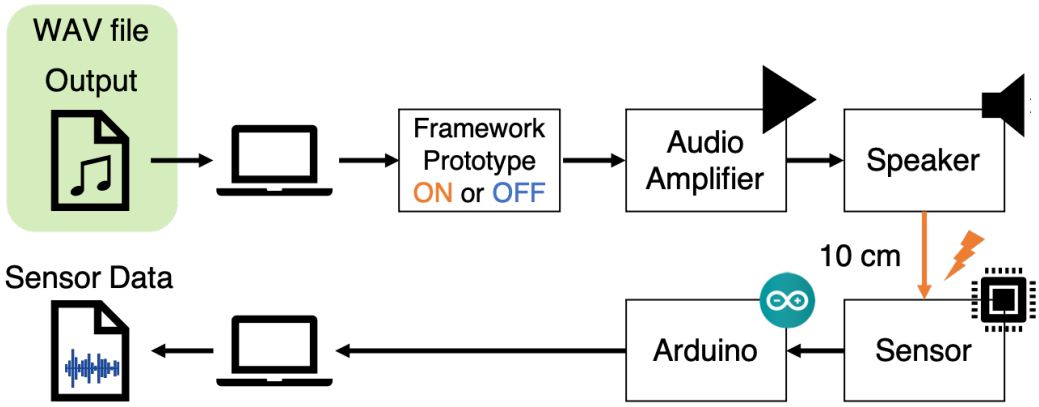


Fig. 3.16 Resonant signal injection attack setup.

read via an Arduino [87] connected to MPU9250. The sound pressure level of the emitted sound waves was adjusted to an average of 100 dB, following the setup used in the previous studies [7, 5]. Next, we identified the resonant frequency of the sensor by emitting sound waves at various frequencies. Resonance was observed at frequencies of 5.2–5.8, 14.0–14.1, 20.25–20.6, 21.3–21.95, and 22.15–22.6 kHz.

Figure 3.17 presents the result for an experiment targeting the frequency of 5650 Hz. In this setup, the Y-axis component exhibited the highest resonance. As indicated by the orange line, the resonance phenomenon that occurred in the Y-axis was successfully restricted, and the sensor value was stabilized. It is thus possible to prevent the resonance attack to the sensor in advance by applying the CPFW framework.

3.7 Discussion

3.7.1 Evaluation Using Generic Analog Signals & Secondary Effect

As we presented in Section 3.3, the design specification of the CPFW framework is generic; it is designed to target any type of analog signals that can be represented in a sequence of data. As an example of an analog signal that has a large impact on CPS security threats, we adopted the audio signal for our experiments. Evaluation

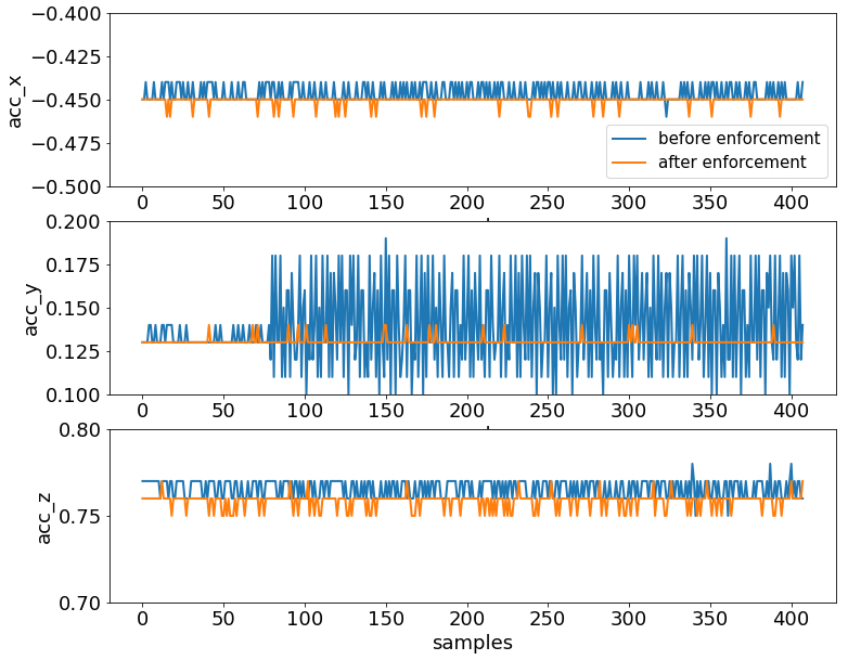


Fig. 3.17 Policy enforcement for resonant attack at 5,650 Hz. Each panel corresponds to an axis (X, Y, Z). The blue line shows the case where sine waves were emitted as the resonance attack, and the orange line shows the case where the sensor value was recorded after the sound waves with frequencies above 4 kHz were filtered out.

for other analog signal output is left for our future study. We note that analog signals exhibit the "secondary effects. " For example, the heat or vibration generated by an actuator that outputs sound or light can be an unexpected input to other CPS devices. Since the characteristics of secondary effects vary from device to device and are often affected by the environment, it is difficult to predict their behavior accurately [88]. A study of access control mechanisms that consider the effects of secondary effects is a future challenge.

3.7.2 Limitation of the CPFW framework

When controlling analog signals, there are cases where feedback control, such as PID control or Kalman filter, is effective, as they are used in various sensor-based systems.

Since feedback control is a method based on past measurements, it introduces an overhead necessary for state management. The CPFW framework does not support the feedback control at present. This limitation can be resolved by employing a module that performs feedback control inside the framework. Evaluation of the effectiveness of such an approach is a topic for future work.

3.7.3 Statistical Anomaly Detection Model & Machine Learning Model

In this paper, we adopt basic statistical values for attributes to detect threat signals. In addition to the attributes illustrated in this paper, anomaly detection score can also be adopted as an attribute. For example, in the anomaly detection **ChangeFinder** [89] using the auto regressive model, the change point score is used to detect changes. It can be used as an attribute by applying it to the frequency statistical attributes. Furthermore, the score obtained from machine learning models [90, 91] can also be used as attributes. These methods compromise real-time performance for basic statistical value. The trade-off between the complexity of the attribute and the real-time performance, in the case of machine learning and anomaly detection at the output side, is a subject for future work.

3.7.4 Ingress vs. Egress Access Control

We mainly focused on egress access control in our experiment. We raised the problem that few devices have countermeasures on the output side and proposed a framework as a mechanism to make it easier to introduce egress access control. The primary objective of this research was to increase the number of devices that implement output-side countermeasures, and reduce the number of options for attackers. If the countermeasure is not possible on the output side, such as custom-develop hardware, then we will use in combination the conventional ingress access control shown in Section 3.8.2.

3.7.5 Legal regulation of analog signal output

Legal systems for transmitting radio signals have been developed in many countries, and government agencies have introduced licensing systems. The regulation of wireless signals is intended to ensure that the shared resource of the radio spectrum is operated safely and to prevent users from transmitting radio waves that could interfere with other communication systems. Such regulations help to prevent misuse of the airwaves, such as jamming attacks and spoofing. As threats exist such as spoofing and DoS in the analog signals generated by CPS devices, regulating the output power is a promising defense measure, similar to radio waves. In addition, it would be effective for companies that manufacture CPS devices to regulate the bandwidth and output power of the interfaces that generate analog signals through a licensing system based on appropriate laws. A combination of litigation with the CPFW framework could regulate the misuse of devices capable of generating ultrasonic waves at extremely high intensities.

3.7.6 Proposal for Policy Sharing System

We have presented some preset policies in this study. Policy diagrams are designed to allow developers and users to develop their own policies. We believe that sharing new policies defined by users and developers is useful. Future work may include providing a mechanism for users to share and reuse such policies. A good starting point is to look at similar community-based rule-sharing schemes such as the community ruleset of Snort [92].

3.8 Related Work

In this section, we review several related works and clarify the advantages of our framework.

3.8.1 Analog Signal Injection Attack

There have been a few research papers aimed at building a generic defending mechanism against the threat posed by malicious analog signal inputs to sensors. Giechaskiel et al. proposed a framework to prevent signal injection attacks [21]. Yan et al. proposed a scheme to formulate injection attacks on sensors [23]. The significant difference between our study and these studies is that we developed a generic and extensible policy-based access control framework that includes egress and ingress access control, while prior studies focus only on ingress access control.

3.8.2 Policy Frameworks for Physical Attacks to IoT Device

Some policy frameworks for monitoring physical interactions have been proposed to address threats caused by physical interactions between internet-of-things devices. Ding et al. defined IoTMon [20], a framework for discovering physical interaction chains that can arise in applications such as IFTTT [67]. Celik et al. created IoTGUARD [93] as a framework for API-based blocking of these physical interactions. To eliminate the errors on the ingress side countermeasures due to noise in the environment, we have extended it to the framework for egress access control in addition to the existing ingress access control.

Although the scope of these studies is different from our study, we believe that their findings help address the "secondary effect" discussed in Section 3.7.1.

3.8.3 AR Input & Output Security

Threats to the sensor input of AR devices have been recognized in previous research [94]. Jana et al. have developed a Recognizer OS that protects the privacy of sensor information and does not give developers information that is not needed for the application [30].

The secure framework named Arya has been proposed to prevent threats from AR output [34, 95]. The AR policy defined in Arya eliminates the AR output threats

to people and provides safe AR Frames. Since AR digital objects have attributes, there is no need to create a specific function to obtain attributes. In our study, we defined attributes to analog signals that do not have explicit attributes. In addition, AR Output Security protects against threats to humans. We extended the scope of threats from humans to devices that target CPS devices. The output of devices will cause security threats not only to people but also to nearby devices. Our research is positioned as the first research to prevent the effects of analog output.

3.9 Summary

We have developed the CPFW framework, a generic and flexible access control mechanism for malicious analog signals targeting CPS devices. The uniqueness of this framework is that it supports both egress and ingress access control mechanisms. This innovation solves the problems of existing ingress access control-based approaches for analog signals, such as degradation of policy violation detection accuracy due to physical noise and difficulty of attack detection due to the nonlinearity of data processing circuits. We also conducted experiments using a prototype of the CPFW framework and demonstrated that it is possible to achieve practical performance. We also demonstrated that the framework could be applied to prevent attacks using malicious analog signals against sensors in CPS devices, e.g., DolphinAttack, adversarial audio example, and WALNUT. Further experiments using general analog signals other than audio signals, developments of more advanced policy violation detection techniques, and building an effective policy sharing scheme for the CPFW framework are left for future challenges.

Chapter 4

Discussion

The limitations and future directions of this study are summarized below.

4.1 Limitations

4.1.1 Evaluation Using Generic Analog Signals

As an example of an analog signal that has a large impact on security threats, we adopted the audio signal for our experiments. Evaluation of threats and countermeasures for other analog signal is left for our future study. It is possible to prevent many analog signal threats in physical space that were not addressed in this thesis, by using our design for light, motor, and wireless signals. We note that analog signals exhibit the "secondary effects. " For example, the heat or vibration generated by an actuator that outputs sound or light can be an unexpected input to other CPS devices. Since the characteristics of secondary effects vary from device to device and are often affected by the environment, it is difficult to predict their behavior accurately [88].

4.1.2 Analog signal transmission in an obstructed environment

The threats verified in this study are based on the assumption that there are no

obstacles in the path of the analog signal. In addition to the countermeasures presented in our study, another option is to physically cover the sensor since most threats cannot be concluded when there is an obstacle between the signal and the sensor. However, this is not a comprehensive countermeasure because biometric sensors in wearable devices may not be able to measure without direct contact with the skin. Light can pass through transparent obstacles such as glass to reach the sensor as an exception. Even in this case, it is necessary to know where the sensor is located in the room [6].

4.1.3 Multi Factor Authentication

This thesis assumed that all authentication systems are single factor authentication, and verified threat models, conducted experiments, and proposed countermeasure models. In cases where a security threat arises against a single sensor, the threat may be mitigated by introducing multi-factor authentication. We did not discuss multi-factor authentication in this study because only mitigating the threats does not lead to a fundamental solution and the number of targets to be discussed is massive due to the combination. The discussion of multi factor authentication is left as future work.

4.2 Future Directions

4.2.1 Physical Security

This study presents the first security threat using the physical phenomena of sound nonlinearity. In the future, security threats or security countermeasure methods based on physics phenomena are expected to increase in response to technological developments using physics phenomena, such as quantum computers. For example, the qubits used in a quantum computer do not use a threshold value to determine the value of a bit, but use the analog value directly, making them less resistant to noise than conventional computers [96]. The noise injection attack shown in this study

may be performed on quantum computers in the future. In addition to the above possibilities, collaborative research between physicists and security researchers may encourage novel and meaningful research.

4.2.2 Safety Engineering for robots

This thesis describes countermeasures against the possibility of analog signals unintentionally harming sensors. The standards and laws in the field of safety engineering, such as the Three Principles of Robotics, are all designed to "prevent harm to humans," and the possibility of a robot causing harm to other robots or IoT devices by its actuators is not a concern of the safety engineering field [97, 98]. For example, ISO 12100 specifies a standard for safety evaluation of systems, but its content is limited to human bodily harm and health risks [98]. In the field of safety engineering, it will be necessary for the future to discuss standards to prevent equipment from creating hazards for other sensors and IoT devices.

4.2.3 Security & Privacy of the Bio Signals

As the number of wearable devices that acquire signals from the human body increases, security threats targeting bio-signals are expected to increase. As the use of bio-signals becomes more widespread, their application to advertising technology is also considered [99]. Bio-signals should be protected with privacy countermeasures like other personal information.

In previous research, the acquisition of biometric signals has only been threatened by direct access to the sensors or by malware to obtain the sensor values. Recently, however, methods for estimating biometric signals using the information other than biometric sensors and generating false signals have been explored [17]. Remote estimation techniques for biometric signals, such as estimation of pulse wave signals by video, were initially studied to reduce the cost of sensors and for convenience [100, 101]. For example, remote Photoplethysmography (rPPG) is a technique for estimating PPG signals from information other than PPG sensors. As a spoofing attack using the rPPG technique, [17] uses CHROM [101], which estimates the PPG signal using the target's video, to trick the system into using PPG

authentication. Similar techniques may be proposed to estimate voice information and EEG signals from appearance information in the future. It is necessary to discuss what can be inferred from the movie, image, and voice information.

Chapter 5

Conclusion

To solve security threats posed by analog signals, as described in Chapter 1, the security threats are clarified using voice as a representative example, and a new approach to countermeasures is proposed to reduce the total number of threats by introducing a new approach to control at the "output side".

In Chapter 2, we focused on sound signals as a threat posed by analog signals and reveal the world's first example of a security threat that takes advantage of nonlinearity in the air, i.e., physical phenomena. Its feasibility was evaluated through extensive user studies and reproducible experiments. We demonstrated that when directional sounds are emitted from parametric loudspeakers and not perceived by a nearby person, attacks can succeed over relatively long distances (2–4 m in a small room and up to 10+ m in a hallway); further, these attacks are tolerant against environmental noises. Although the Audio Hotspot Attack is currently a proof-of-concept, possible countermeasures to render the threats unsuccessful have been provided. The proposed attack uses ultrasound self-demodulation, which is a parametric phenomenon.

In Chapter 3, we developed the CPFW framework, a generic and flexible access control mechanism for malicious analog signals targeting CPS devices. The uniqueness of this framework is that it supports both egress and ingress access control mechanisms. This innovation solves the problems of existing ingress access control-based approaches for analog signals, such as degradation of policy violation

detection accuracy due to physical noise and difficulty of attack detection due to the nonlinearity of data processing circuits. We also conducted experiments using a prototype of the CPFW framework and demonstrated that it is possible to achieve practical performance. We also demonstrated that the framework could be applied to prevent attacks using malicious analog signals against sensors in CPS devices, e.g., DolphinAttack, adversarial audio example, and WALNUT.

The advantages of this doctoral thesis are that (1) it classifies threats caused by analog signals and identifies threats caused by physical phenomena for the first time, (2) it implements countermeasures to detect threat signals and identifies problems commonly found in the countermeasures, and (3) it presents a framework to analyze potential threat analog signals at the output side to solve problems found in (2). The contribution of this doctoral thesis has led to the development of a new academic field of security and privacy of biometric signals and physical security, which is being developed by the knowledge of physics. This work is the first step in research to solve security and privacy problems at the interface between physics and informatics. In terms of contributions to society, we reported our findings of threats to physical analog signals to Goodle and LINE, and asked them to make improvements. We have summarized our presentation in IEEE spectrum to alert the public. Furthermore, by presenting the design of the framework, we were able to provide a guideline for future standardization of analog signals. We believe that we have provided an opportunity to stimulate discussion and design of what should be required to make analog signals and outputs safe. We hope that this research will further develop analog signal security and physical security.

Acknowledgement

Firstly, I would like to express my deepest gratitude to my supervisor Prof. Tatsuya Mori for his passionate guidance, encouragement, and immense knowledge. His world-class research ability always helps our research. Under his research guidance, I have learned the importance of making a plan and carrying it out. I would especially like to thank his constant support of my research.

I would like to thank Prof. Tetsuji Ogawa, sub-chief referee of my dissertion, he advice me about how to proceed our research. I would like to thank co-writer of my paper, advisor, and referee, Prof.Yasuhiro Oikawa. He advice our resesarch in terms of the audio signal. He not only worked with me to plan and study the sound signal research, but also held meetings to discuss the experimental design, and helped us prepare the equipment. Prof. Takeshi Sugawara, one of the referee of my dissertion, was interested in our research concept of "output security" and discuss with us. Thanks to your contact, we can get motivated to our research more. I alto thank Syota Minami and Zhou Yunao, who are the co-writers of my research. They help experiments setup and knowledge around acoustics and hardware fields. Shota Minami helped our experiment when I wanted to do experiment during New Year's. I also thank Tatsuya Takehisa and Takeshi Takahashi, in National Institute of Information and Communications Technology as discussion members for my research. I would like to thank all the members of Network Security Laboratory for their ideas and support. Most of all, I would like to thank my friends and family for their warm support. I would especially like to thank my father and mother for their constant support of my research. Finally, I would like to thank those who read and commented on this paper; Tatsuya Takehisa, Miki Yamazaki, Taiga Ono.

Bibliography

[1] Daniel Harris and Marius Miknis. Fault-tolerant iot cloud orchestrator, 09 2020.

[2] Ruud Bolle, et al. *Biometrics, Personal Identification in Networked Society: Personal Identification in Networked Society*. Kluwer Academic Publishers, USA, 1998.

[3] Nicholas Carlini and David A. Wagner. Audio adversarial examples: Targeted attacks on speech-to-text. *CoRR*, 2018.

[4] Md. Sahidullah, et al. Introduction to voice presentation attack detection and recent advances. In *Computational Intelligence in Electromyography Analysis - A Perspective on Current Applications and Future Challenges*, pp. 321–361. 2019.

[5] Yunmok Son, Hocheol Shin, Dongkwan Kim, Young-Seok Park, Juhwan Noh, Kibum Choi, Jungwoo Choi, and Yongdae Kim. Rocking drones with intentional sound noise on gyroscopic sensors. In *the 24th USENIX Security Symposium*, pp. 881–896, 2015.

[6] Takeshi Sugawara, Benjamin Cyr, Sara Rampazzi, Daniel Genkin, and Kevin Fu. Light commands: Laser-based audio injection attacks on voice-controllable systems. In *the 29th USENIX Security Symposium*, 2020.

[7] Timothy Trippel, Ofir Weisse, Wenyuan Xu, Peter Honeyman, and Kevin Fu. WALNUT: waging doubt on the integrity of MEMS accelerometers with acoustic injection attacks. In *2017 IEEE European Symposium on Security and Privacy, EuroS&P*, pp. 3–18. IEEE, 2017.

[8] Guoming Zhang, et al. Dolphinattack: Inaudible voice commands. In *Proceedings of the 2017 ACM SIGSAC, CCS*, pp. 103–117, 2017.

[9] Tavish Vaidya, et al. Cocaine noodles: Exploiting the gap between human and machine speech recognition. In *9th USENIX Workshop on Offensive Technologies, WOOT*, 2015.

[10] Xiaoyu Ji, Juchuan Zhang, Shui Jiang, Jishen Li, and Wenyuan Xu. Capspeaker: Injecting voices to microphones via capacitors. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, CCS '21, p. 1915–1929, 2021.

[11] Yuxin Chen, Huiying Li, Shan-Yuan Teng, Steven Nagels, Zhijing Li, Pedro Lopes, Ben Y. Zhao, and Haitao Zheng. Wearable microphone jamming. In *ACM SIGCHI Conference on Human Factors in Computing Systems, CHI*, pp. 1–12, 2020.

[12] Dibya Mukhopadhyay, Maliheh Shirvanian, and Nitesh Saxena. All your voices are belong to us: Stealing voices to fool humans and machines. In *Computer Security - ESORICS 2015 - 20th European Symposium on Research in Computer Security, Vienna, Austria, September 21-25, 2015, Proceedings, Part II*, pp. 599–621, 2015.

[13] Tomi Kinnunen, et al. The asvspoof 2017 challenge: Assessing the limits of replay spoofing attack detection. In Francisco Lacerda, editor, *Proceedings of INTERSPEECH 2017, 18th Annual Conference of the International Speech Communication Association, Stockholm, Sweden, August 20-24, 2017*, pp. 2–6. ISCA, 2017.

[14] Nitesh Saxena Dibya Mukhopadhyay, Maliheh Shirvanian. All your voices are belong to us: Stealing voices to fool humans and machines. In *In Proceedings of the European Symposium on Research in Computer Security.*, Springer, pp. 599–621, 2015.

[15] Nicholas Carlini, Pratyush Mishra, Tavish Vaidya, Yuankai Zhang, Micah Sherr, Clay Shields, David A. Wagner, and Wenchao Zhou. Hidden voice commands. In *Proceedings of 25th USENIX Security Symposium*, pp. 513–530, 2016.

[16] Simon Eberz, Nicola Paoletti, Marc Roeschlin, Andrea Patané, Marta Kwiatkowska, and Ivan Martinovic. Broken hearted: How to attack ecg

biometrics. In *24th Annual Network and Distributed System Security Symposium, NDSS 2017, San Diego, California, USA, February 26 - March 1, 2017*. The Internet Society, 2017.

[17] Lin Li, Chao Chen, Lei Pan, Jun Zhang, and Yang Xiang. Video is all you need: Attacking ppg-based biometric authentication, 2022.

[18] D Jude Hemanth, J Anitha, and George A Tsihrintzis. *Internet of medical things : remote healthcare systems and applications*. Springer, 2021.

[19] Yulong Cao, Ningfei Wang, Chaowei Xiao, Dawei Yang, Jin Fang, Ruigang Yang, Qi Alfred Chen, Mingyan Liu, and Bo Li. Invisible for both camera and lidar: Security of multi-sensor fusion based perception in autonomous driving under physical-world attacks. In *42nd IEEE Symposium on Security and Privacy, SP 2021, San Francisco, CA, USA, 24-27 May 2021*, pp. 176–194. IEEE, 2021.

[20] Wenbo Ding and Hongxin Hu. On the safety of iot device physical interaction control. In *The 25th ACM SIGSAC Conference on Computer and Communications Security, CCS*, pp. 832–846, 2018.

[21] Ilias Giechaskiel, Youqian Zhang, and Kasper Bonne Rasmussen. A framework for evaluating security in the presence of signal injection attacks. In *Computer Security - ESORICS 2019*, Vol. 11735 of *Lecture Notes in Computer Science*, pp. 512–532. Springer, 2019.

[22] Andreas Nautsch, Xin Wang, Nicholas W. D. Evans, Tomi H. Kinnunen, Ville Vestman, Massimiliano Todisco, Héctor Delgado, Md. Sahidullah, Junichi Yamagishi, and Kong Aik Lee. Asvspoof 2019: Spoofing countermeasures for the detection of synthesized, converted and replayed speech. *IEEE Trans. Biom. Behav. Identity Sci.*, Vol. 3, No. 2, pp. 252–265, 2021.

[23] Chen Yan, Hocheol Shin, Connor Bolton, Wenyuan Xu, Yongdae Kim, and Kevin Fu. Sok: A minimalist approach to formalizing analog sensor security. In *2020 IEEE Symposium on Security and Privacy, S&P 2020*, pp. 233–248. IEEE, 2020.

[24] Sayaka Shiota, et al. Voice liveness detection algorithms based on pop noise caused by human breath for automatic speaker verification. In *INTERSPEECH*

2015, 16th Annual Conference of the International Speech Communication Association, Dresden, Germany, September 6-10, 2015 [102], pp. 239–243.

[25] Massimiliano Todisco, Héctor Delgado, and Nicholas W. D. Evans. A new feature for automatic speaker verification anti-spoofing: Constant Q cepstral coefficients. In Luis Javier Rodríguez-Fuentes and Eduardo Lleida, editors, *Odyssey 2016: The Speaker and Language Recognition Workshop, Bilbao, Spain, June 21-24, 2016*, pp. 283–290. ISCA, 2016.

[26] Xiong Xiao, et al. Spoofing speech detection using high dimensional magnitude and phase features: the NTU approach for asvspoof 2015 challenge. In *Proceedings of INTERSPEECH 2015, 16th Annual Conference of the International Speech Communication Association* [102], pp. 2052–2056.

[27] Galina Lavrentyeva, et al. Audio replay attack detection with deep learning frameworks. In Francisco Lacerda, editor, *Interspeech 2017, 18th Annual Conference of the International Speech Communication Association*, pp. 82–86. ISCA, 2017.

[28] Marcin Witkowski, Stanislaw Kacprzak, Piotr Żelasko, Konrad Kowalczyk, and Jakub Galka. Audio replay attack detection using high-frequency features. In *INTERSPEECH*, 2017.

[29] Xiaohai Tian, Xiong Xiao, Eng Siong Chng, and Haizhou Li. Spoofing speech detection using temporal convolutional neural network. In *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference, APSIPA*, pp. 1–6. IEEE, 2016.

[30] Suman Jana, David Molnar, Alexander Moshchuk, Alan M. Dunn, Benjamin Livshits, Helen J. Wang, and Eyal Ofek. Enabling fine-grained permissions for augmented reality applications with recognizers. In *the 22th USENIX Security Symposium*, pp. 415–430, 2013.

[31] Pavel Korshunov and Sébastien Marcel. A cross-database study of voice presentation attack detection. In *Handbook of Biometric Anti-Spoofing - Presentation Attack Detection, Second Edition*, pp. 363–389. 2019.

[32] Denis Foo Kune, et al. Ghost talk: Mitigating EMI signal injection attacks against analog sensors. In *Proceedings of 2013 IEEE Symposium on Security*

and Privacy, SP, pp. 145–159. IEEE Computer Society, 2013.

[33] Nirupam Roy, Sheng Shen, Haitham Hassanieh, and Romit Roy Choudhury. Inaudible voice commands: The long-range attack and defense. In *15th USENIX, NSDI 2018*, pp. 547–560. USENIX Association, 2018.

[34] Kiron Lebeck, Kimberly Ruth, Tadayoshi Kohno, and Franziska Roesner. Securing augmented reality output. In *2017 IEEE Symposium on Security and Privacy, S&P*, pp. 320–337. IEEE Computer Society, 2017.

[35] Apple. ios - siri, 2018.

[36] Google. google-assistant, 2018.

[37] Amazon. Amazon alexa, 2018.

[38] M. Yoneyama, et al. The audio spotlight: An application of nonlinear interaction of sound waves to a new type of loudspeaker design. *The Journal of the Acoustical Society of America*, Vol. 73, No. 5, pp. 1532–1536, 1983.

[39] Peter J Westervelt. Parametric acoustic array. *The Journal of the Acoustical Society of America*, Vol. 35, No. 4, pp. 535–537, 1963.

[40] Woon-Seng Gan, et al. A review of parametric acoustic array in air. *Applied Acoustics*, Vol. 73, No. 12, pp. 1211 – 1219, 2012. Parametric Acoustic Array: Theory, Advancement and Applications.

[41] S. N. Gurbatov, O. V. Rudenko, and A. I. Saichev. *Waves and Structures in Nonlinear Nondispersive Media [electronic resource] : General Theory and Applications to Nonlinear Acoustics*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2nd. edition, 2012.

[42] International Organization for Standardization. *ISO/IEC 30107. Information technology – biometric presentation attack detection.* International Organization for Standardization, 2016.

[43] Rosa González Hautamäki, Tomi Kinnunen, Ville Hautamäki, Timo Leino, and Anne-Maria Laukkanen. I-vectors meet imitators: on vulnerability of speaker verification systems against voice mimicry. In *Proceedings of INTERSPEECH 2013, 14th Annual Conference of the International Speech Communication Association, Lyon, France, August 25-29, 2013*, pp. 930–934, 2013.

[44] Zhizheng Wu, et al. A study on replay attack and anti-spoofing for text-dependent speaker verification. In *Proceedings of Asia-Pacific Signal and Information Processing Association Annual Summit and Conference, APSIPA*, pp. 1–5, 2014.

[45] Aäron van den Oord, et al. Wavenet: A generative model for raw audio. In *The 9th ISCA Speech Synthesis Workshop*, p. 125, 2016.

[46] eMarketer. Amazon echo losing share as speaker rivalry heats up, 2018.

[47] switchscience. Super directional speaker kit, 2017.

[48] Accuphase. Acouspade, 2001.

[49] Ultrasonic Audio Technologies. Acouspade, 2018.

[50] YAMAHA. Monitor speaker ms101 iii owner's manual, 2018.

[51] RION. The nl series line up, 2018.

[52] B&K. Product data: Teds microphones (bp2225), 2018.

[53] motu. Ultralitemk4 overview, 2018.

[54] Amazon. Amazon polly, 2018.

[55] Center for Hearing and Communication. Common environmental noise levels, 2018.

[56] Thomas D. Rossing. *Springer Handbook of Acoustics*. Springer, 2 edition, 2014.

[57] Stéphane Pigeon. Babble noise -frequency-shaped babble noise generator, 2019.

[58] MathWorks. Matlab, 1994–2019.

[59] Simon Grondin. *Psychology of Perception*. Springer, 2016.

[60] Linghan Zhang, et al. 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, CCS*, pp. 57–71. ACM, 2017.

[61] Nirupam Roy, et al. Backdoor: Making microphones hear inaudible sounds. In *Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services, MobiSys'17*, pp. 2–14, 2017.

[62] Linghan Zhang, et al. Voicelive: A phoneme localization based liveness

detection for voice authentication on smartphones. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pp. 1080–1091, 2016.

[63] Sayaka Shiota, et al. Voice liveness detection for speaker verification based on a tandem single/double-channel pop noise detector. In *Odyssey 2016: The Speaker and Language Recognition Workshop*, pp. 259–263, 2016.

[64] Nicholas Carlini and David A. Wagner. Audio adversarial examples: Targeted attacks on speech-to-text. In *2018 IEEE Security and Privacy Workshops, SP Workshops 2018, San Francisco, CA, USA, May 24, 2018*, pp. 1–7. IEEE, 2018.

[65] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian J. Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *CoRR*, Vol. abs/1312.6199, , 2013.

[66] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *CoRR*, Vol. abs/1412.6572, , 2014.

[67] IFTTT Platform. Ifttt, 2018.

[68] Ryo Iijima, Shota Minami, Yunao Zhou, Tatsuya Takehisa, Takeshi Takahashi, Yasuhiro Oikawa, and Tatsuya 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, 2021.

[69] Man Zhou, Zhan Qin, Xiu Lin, Shengshan Hu, Qian Wang, and Kui Ren. Hidden voice commands: Attacks and defenses on the VCS of autonomous driving cars. *IEEE Wirel. Commun.*, Vol. 26, No. 5, pp. 128–133, 2019.

[70] GNU Radio, 2021.

[71] Angkoon Phinyomark, Sirinee Thongpanja, Huosheng Hu, Pornchai Phukpattaranont, and Chusak Limsakul. The usefulness of mean and median frequencies in electromyography analysis. In Ganesh R. Naik, editor, *Computational Intelligence in Electromyography Analysis*, chapter 8. IntechOpen, Rijeka, 2012.

[72] Rajesh Bachu, S. Kopparthi, B. Adapa, and Buket D. Barkana. Voiced/un-

voiced decision for speech signals based on zero-crossing rate and energy. In *Advanced Techniques in Computing Sciences and Software Engineering, Volume II of the proceedings of the 2008 International Conference on SCSS*, pp. 279–282. Springer, 2008.

[73] Dong Enqing, Liu Guizhong, Zhou Yatong, and Cai Yu. Voice activity detection based on short-time energy and noise spectrum adaptation. In *6th International Conference on Signal Processing, 2002.*, Vol. 1, pp. 464–467 vol.1, 2002.

[74] Google. Speech-to-text accurately convert speech into text using an api powered by google' s ai technologies., 2021.

[75] Tim Sainburg, Marvin Thielk, and Timothy Q Gentner. Finding, visualizing, and quantifying latent structure across diverse animal vocal repertoires. *PLoS computational biology*, Vol. 16, No. 10, p. e1008228, 2020.

[76] Simulink Documentation. Simulation and model-based design, 2020.

[77] Daniel Arp, Erwin Quiring, Christian Wressnegger, and Konrad Rieck. Privacy threats through ultrasonic side channels on mobile devices. In *Proceedings of 2017 IEEE European Symposium on Security and Privacy, EuroSP 2017, Paris, France, April 26-28, 2017*, pp. 35–47. IEEE, 2017.

[78] Raspberry Pi Foundation. Raspberry pi os lite release notes, 2021.

[79] Google. Aiy voice kit, 2021.

[80] Raspberry Pi Foundation. Raspberry pi 3b+, 2021.

[81] ITU-T. G.114 : One-way transmission time, 2003. `https://www.itu.int/rec/T-REC-G.114-200305-I/en`.

[82] Michael Stone, Brian Moore, Katrin Meisenbacher, and Peter Derleth. Tolerable hearing aid delays. v. estimation of limits for open canal fittings. *Ear and hearing*, Vol. 29, pp. 601–17, 09 2008.

[83] Dario Amodei, et al. Deep speech 2 : End-to-end speech recognition in english and mandarin. In Maria-Florina Balcan and Kilian Q. Weinberger, editors, *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016*, Vol. 48 of *JMLR Workshop and Conference Proceedings*, pp. 173–182. JMLR.org, 2016.

[84] InvenSense Inc. Tdk-invensense motion sensor universal evaluation board (uevb) user guide, 2014. https://invensense.tdk.com/download-pdf/invensense-motion-sensor-universal-evaluation-board-uevb-user-guide/.

[85] Pyramid. Pyramid tw28 3.75" aluminum bullet horn tweeter pair with swivel housing, 2021.

[86] YAMAHA Inc. A-s501 integrated amplifier, 2021. https://usa.yamaha.com/products/audio_visual/hifi_components/a-s501/downloads.html.

[87] Arduino. Arduino, 2021.

[88] Bonfiglioli Riduttori, et al. *Gear Motor Handbook*. Springer Berlin Heidelberg, Berlin, Heidelberg, 1st ed. edition, 1995.

[89] shunsuke aihara. Github: shunsukeaihara / changefinder, 2021.

[90] Chunlei Zhang, Chengzhu Yu, and John H. L. Hansen. An investigation of deep-learning frameworks for speaker verification antispoofing. *J. Sel. Topics Signal Processing*, Vol. 11, No. 4, pp. 684–694, 2017.

[91] Benedikt Eiteneuer and Oliver Niggemann. LSTM for model-based anomaly detection in cyber-physical systems. *CoRR*, Vol. abs/2010.15680, , 2020.

[92] Martin Roesch. Snort - lightweight intrusion detection for networks. In *the 13th USENIX Conference*, LISA '99, p. 229–238, 1999.

[93] Z. Berkay Celik, Gang Tan, and Patrick D. McDaniel. Iotguard: Dynamic enforcement of security and safety policy in commodity iot. In *26th Annual Network and Distributed System Security Symposium, NDSS*. The Internet Society, 2019.

[94] Kiron Lebeck, Tadayoshi Kohno, and Franziska Roesner. How to safely augment reality: Challenges and directions. In *Proceedings of the 17th International Workshop on Mobile Computing Systems and Applications*, HotMobile '16, p. 45–50. Association for Computing Machinery, 2016.

[95] Kiron Lebeck, Kimberly Ruth, Tadayoshi Kohno, and Franziska Roesner. Arya: Operating system support for securely augmenting reality. *IEEE Secur. Priv.*, Vol. 16, No. 1, pp. 44–53, 2018.

[96] Abdullah Ash Saki, Mahabubul Alam, and Swaroop Ghosh. Impact of noise on the resilience and the security of quantum computing. In *2021 22nd International Symposium on Quality Electronic Design (ISQED)*, pp. 186–191, 2021.

[97] Nancy Leveson. System safety engineering: Back to the future. 01 2008.

[98] International Organization for Standardization. ISO 12100:2010. Safety of machinery — General principles for design — Risk assessment and risk reduction. International Organization for Standardization, 2010.

[99] Rahmadi Trimananda, Hieu Le, Hao Cui, Janice Tran Ho, Anastasia Shuba, and Athina Markopoulou. Ovrseen: Auditing network traffic and privacy policies in oculus vr, 2021.

[100] Dwaipayan Biswas, Luke Everson, Muqing Liu, Madhuri Panwar, Bram-Ernst Verhoef, Shrishail Patki, Chris H. Kim, Amit Acharyya, Chris Van Hoof, Mario Konijnenburg, and Nick Van Helleputte. Cornet: Deep learning framework for ppg-based heart rate estimation and biometric identification in ambulant environment. *IEEE Transactions on Biomedical Circuits and Systems*, Vol. 13, No. 2, pp. 282–291, 2019.

[101] G. de Haan and V. Jeanne. Robust pulse rate from chrominance-based rppg. *IEEE Transactions on Biomedical Engineering*, Vol. 60, No. 10, pp. 2878–2886, 2013.

[102] *INTERSPEECH 2015, 16th Annual Conference of the International Speech Communication Association, Dresden, Germany, September 6-10, 2015*. ISCA, 2015.

List of Research Achievements

Journal Papers (Reviewed)

1. Ryo Iijima, Shota Minami, Yunao Zhou, Tatsuya Takehisa, Takeshi Takahashi, Yasuhiro Oikawa, Tatsuya, Mori, "Audio Hotspot Attack: An Attack on Voice Assistance Systems Using Directional Sound Beams and its Feasibility," IEEE Transactions on Emerging Topics in Computing PP(99):2004–2018, Volume: 9, Issue: 4, Oct.-Dec. 1, 2021 (online access: https://ieeexplore.ieee.org/document/8906174)

Conference Papers (Reviewed)

1. Ryo Iijima, Tatsuya Takehisa, and Tatsuya Mori, "Cyber-Physical Firewall: Mitigating the Threats Caused by Malicious Analog Signals", Malicious Software and Hardware in Internet of Things, Proceedings of the ACM International Conference on Computing Frontiers 2022, pp.296–304, 2022
2. Atsuko Natatsuka, Ryo Iijima, Takuya Watanabe, Mitsuaki Akiyama, Tetsuya Sakai, Tatsuya Mori, "Understanding the Behavior Transparency of Voice Assistant Applications Using the ChatterBox Framework," Proceedings of the 25th International Symposium on Research in Attacks, Intrusions and Defenses (RAID 2022), October 2022

Posters (Reviewed)

1. Ryo Iijima, Shota Minami, Yunao Zhou, Tatsuya Takehisa, Takeshi Takahashi, Yasuhiro Oikawa, Tatsuya, Mori, "POSTER: Audio Hotspot Attack: An Attack on Voice Assistance Systems Using Directional Sound Beams"

(Poster Presentation), CCS '18 Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security pp.2222-2224 Oct 2018

2. Atsuko Natatsuka, Ryo Iijima, Takuya Watanabe, Mitsuaki Akiyama, Tetsuya Sakai, Tatsuya Mori, "Poster: A First Look at the Privacy Risks of Voice Assistant Apps," Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security, pp.2633-2635, Nov 2019

Invited Talks (Oral Presentation)

1. Ryo Iijima, "Latest Trends in Voice Assistance Systems and Security Research: An Attack on Voice Assistance Systems Using Directional Sound Beams," The 14th International Workshop on Security (IWSEC) August 2019 【invited talks】
2. 飯島 涼, "音声認識×セキュリティ研究の最新動向 〜 超音波の分離放射による音声認識機器への 攻撃と対策手法の提案〜," ハードウェアセキュリティフォーラム 2018 (HWS2018) 2018 年 12 月 13 日 【招待講演】

Others

1. 飯島 涼, 竹久 達也, 森 達哉, "アナログ信号によるセキュリティ脅威をアナログ信号による脅威を検知・規制するセキュリティフレームワークの提案と検証," コンピュータセキュリティシンポジウム 2021, 2021 年 10 月
2. 飯島 涼, 竹久 達也, 高橋 健志, 森 達哉, "Output Security Framework: アナログ信号によるセキュリティ脅威を出力側で規制するフレームワークの提案," 暗号と情報セキュリティシンポジウム 2021 (SCIS2021) 2021 年 1 月
3. 飯島 涼, 南 翔汰, シュウ インゴウ, 竹久 達也, 高橋 健志, 及川 靖広, 森 達哉, "超音波の分離放射による音声認識機器への攻撃: ユーザスタディ評価と対策技術の提案," コンピュータセキュリティシンポジウム 2018 論文集 17-24 2018 年 10 月
4. 飯島 涼, "指向性スピーカを用いた音声認識装置への攻撃と評価," サイバーセキュリティシンポジウム道後 2018 (SEC 道後 2018) 2018 年 3 月
5. 飯島 涼, 南 翔汰, シュウ インゴウ, 及川 靖広, 森 達哉, "パラメトリックスピーカーを利用した音声認識機器への攻撃と評価" 暗号と情報セキュリ

ティシンポジウム 2018 (SCIS2018) 2018 年 1 月

6. 竹久 達也, 丑丸 逸人, 牧田 大佑, 有末 大, 三村 聡志, 末田 卓巳, 飯島 涼, 伊沢 亮一, 井上 大介,“PPG センサを用いたウェアラブルデバイスに対する偽容積脈波提示攻撃に関する一考察,” コンピュータセキュリティシンポジウム 2021, 2021 年 10 月
7. 南澤 勇太, 飯島 涼, 森 達哉, “慣性計測装置に対する共振誘発攻撃の評価,” 情報通信システムセキュリティ研究会 (ICSS2021), 2021 年 3 月
8. 刀塚 敦子, 飯島 涼, 渡邉 卓弥, 秋山 満昭, 酒井 哲也, 森 達哉, “Voice Assistant アプリの対話型解析システムの開発, ” 情報通信システムセキュリティ研究会 (ICSS2021) 2021 年 3 月
9. 刀塚 敦子, 飯島 涼, 渡邉 卓弥, 秋山 満昭, 酒井 哲也, 森 達哉, “Voice Assistant アプリの大規模実態調査,” コンピュータセキュリティシンポジウム 2019（CSS2019）, 2019 年 10 月
10. 飯島 涼, 南 翔汰, シュウ インゴウ, 及川 靖広, 森 達哉, コンピュータセキュリティシンポジウム 2018（CSS2018） 最優秀論文賞 超音波の分離放射による音声認識機器への攻撃: ユーザスタディ評価と対策技術の提案, 2018 年 10 月
11. 飯島 涼, 南 翔汰, シュウ インゴウ, 及川 靖広, 森 達哉, 暗号と情報セキュリティシンポジウム 2018 (SCIS2018) SCIS 論文賞 パラメトリックスピーカーを利用した音声認識機器への攻撃と評価, 2018 年 4 月
12. 飯島 涼, サイバーセキュリティシンポジウム道後 2018 (SEC 道後 2018) 最優秀学生研究賞 “指向性スピーカを用いた音声認識機器への攻撃と評価”, 2018 年 3 月
13. 飯島 涼, 山下記念研究賞, 2019 年 10 月
14. 刀塚敦子, 飯島 涼, 渡邉卓弥, 秋山満昭, 酒井哲也, 森達哉, コンピュータセキュリティシンポジウム 2019（CSS2019）最優秀論文賞 “Voice Assistant アプリの大規模実態調査”, 2019 年 10 月
15. 飯島 涼, アーリーバードプログラム, 早稲田大学 理工学術院総合研究所, 2021
16. 飯島 涼, 研究助成 A: “生体電位を用いたウェアラブルデバイス向け動作認証方式の開発”, 公益財団法人 立石科学技術振興財団, 2022

17. 飯島 涼，若手研究: “生体電位を用いたウェアラブルデバイス向け動作認証方式の開発”，科学研究費助成事業，日本学術振興会，2022–2024

Copyrights