Using Sonar for Liveness Detection to Protect Smart Speakers against Remote Attackers

YEONJOON LEE, Hanyang University, Republic of Korea
YUE ZHAO, SKLOIS, Institute of Information Engineering, Chinese Academy of Sciences and School of Cyber Security, University of Chinese Academy of Sciences, China
JIUTIAN ZENG, SKLOIS, Institute of Information Engineering, Chinese Academy of Sciences and School of Cyber Security, University of Chinese Academy of Sciences, China
KWANGWUK LEE, Indiana University Bloomington, United States
NAN ZHANG, Indiana University Bloomington, United States
FAYSAL HOSSAIN SHEZAN, University of Virginia, United States
YUAN TIAN, University of Virginia, United States
KAI CHEN, SKLOIS, Institute of Information Engineering, Chinese Academy of Sciences and School of Cyber Security, University of Chinese Academy of Sciences, China
XIAOFENG WANG, Indiana University Bloomington, United States

Smart speakers, which wait for voice commands and complete tasks for users, are becoming part of common households. While voice commands came with basic functionalities in the earlier days, as the market grew, various commands with critical functionalities were developed; e.g., access banking services, send money, open front door. Such voice commands can cause serious consequences once smart speakers are attacked. Recent research shows that smart speakers are vulnerable to malicious voice commands sent from other speakers (e.g., TV, baby monitor, radio) in the same area. In this work, we propose the Speaker-Sonar, a sonar-based liveness detection system for smart speakers. Our approach aims to protect the smart speakers from remote attackers that leverage network-connected speakers to send malicious commands. The key idea of our approach is to make sure that the voice command is indeed coming from the user. For this purpose, the Speaker-Sonar emits an inaudible sound and tracks the user's direction to compare it with the direction of the received voice command. The Speaker-Sonar does not require additional action from the user and works through an automatic consistency check. We built the Speaker-Sonar on a raspberry pi 3b, a circular microphone array, and a commodity speaker by imitating the Amazon Echo. Our evaluation shows that the Speaker-Sonar can reject remote voice attacks with an average accuracy of 95.5% in 2 meters, which significantly raises the bar for remote attackers. To the best of our knowledge, our defense is able to defend against known remote voice attack techniques.

Authors' addresses: Yeonjoon Lee, Hanyang University, Republic of Korea, yeonjoonlee@hanyang.ac.kr; Yue Zhao, SKLOIS, Institute of Information Engineering, Chinese Academy of Sciences, School of Cyber Security, University of Chinese Academy of Sciences, China, zhaoyue@iie.ac.cn; Jiutian Zeng, SKLOIS, Institute of Information Engineering, Chinese Academy of Sciences, School of Cyber Security, University of Chinese Academy of Sciences, China, zengjiutian@iie.ac.cn; Kwangwuk Lee, Indiana University Bloomington, United States, lee2074@indiana.edu; Nan Zhang, Indiana University Bloomington, United States, nz3@indiana.edu; Faysal Hossain Shezan, University of Virginia, United States, faysalhossain2007@gmail.com; Yuan Tian, University of Virginia, United States, yuant@virginia.edu; Kai Chen, SKLOIS, Institute of Information Engineering, Chinese Academy of Sciences, School of Cyber Security, University of Chinese Academy of Sciences, China, chenkai@iie.ac.cn; XiaoFeng Wang, Indiana University Bloomington, United States, xw7@indiana.edu.

2474-9567/2020/3-ART16 $15.00
https://doi.org/10.1145/3380991

CCS Concepts: • **Human-centered computing → Ubiquitous and mobile computing**; Ubiquitous and mobile computing design and evaluation methods.

Additional Key Words and Phrases: sonar, smart speakers, liveness detection

ACM Reference Format:
Yeonjoon Lee, Yue Zhao, Jiutian Zeng, Kwangwuk Lee, Nan Zhang, Faysal Hossain Shezan, Yuan Tian, Kai Chen, and XiaoFeng Wang. 2020. Using Sonar for Liveness Detection to Protect Smart Speakers against Remote Attackers. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 4, 1, Article 16 (March 2020), 28 pages. https://doi.org/10.1145/3380991

1 INTRODUCTION

Smart speakers, also known as intelligent voice assistants such as Amazon Echo and Google Home, are becoming popular; as of May 2018, 54.4 million people in the U.S own a smart speaker [20]. Furthermore, the number of skill (i.e., voice apps) counts in Amazon Alexa surpassed 30,000 in the U.S. [3]. Such devices wait for a user voice command and are triggered by particular keywords, such as "Alexa" and "Hey, Google", and follow the voice command received to perform various activities which include critical services. For example, the smart speakers can help users to transfer money (e.g., Paypal), shop online, access banking services, place phone calls, schedule appointments, check emails, control cars (e.g., learn location, unlock, start engine), and even control smart home devices (e.g., open front door, change security setting of home from away mode to stay mode which disables motion-sensors). Recent articles report that other critical commands are on the way: e.g., Amazon is considering to use Alexa to start a person-to-person payments feature [8]; Amazon pay is going to be available for donations, restaurants and event ticketing [7]. Smart speakers, the interface of such critical voice commands, should be well protected as the consequences can be serious once they are attacked.

Emerging Threats to Smart Speakers. Since the smart speakers are continuously listening and are waiting for voice commands, they can be attacked by malicious voice commands sent from nearby network-connected devices: e.g., audio from television advertisements triggered Amazon Echo to place orders for dollhouse [5], Google Home to describe Whopper burger [11]. In addition, network-connected devices with speakers are emerging and becoming more common in the customer's homes: e.g., smart TV, baby monitor, security camera. Indeed, it has been reported that such devices are vulnerable to attacks and can be hacked: for example, security-critical weaknesses have been found in Smart TV [12], Belkin NetCam [14], baby monitors [19]. Moreover, according to a recent report from Consumer Reports, millions of smart TVs are found to be vulnerable and can be controlled by hackers exploiting easy-to-find security flaws [12]. Once such network-connected devices are hacked, smart speakers become exposed to attacks. Such observations show that a threat against smart speakers becomes increasingly realistic.

In the meantime, recent research demonstrates that smart speakers are vulnerable to various types of voice command attacks which use different techniques: The replay attack [62, 63] and voice synthesizing attack [31, 65] can send malicious voice command that mimics users voice whereas the dolphin attack [70] and inaudible voice commands (long-range version of dolphin attack) [52] are more stealthy, producing commands inaudible to human.

Despite the success of smart speakers and the dramatic growth of its voice app store (e.g., Alexa skills store), little has been done to protect them. Considering the number of smart speaker users and the type of voice apps they offer, once the smart speakers are attacked, the consequences could be serious: e.g., cause financial loss by transferring money to a malicious party, threatening the safety of the users by controlling smart home devices. Moreover, the popularity of network-connected devices which are shown to be often vulnerable [12, 14, 19] and the research on attacks targeting smart speakers warns that protection of such smart speakers is necessary. So far, several defenses have been proposed: e.g., smart speakers have an option to set a pin code. However, this has only been used for purchase. Also, prior work [59] proposes multiple defense options: audio/visual feedback on reception of a command; audio captcha to verify whether the command is coming from a human, etc. EchoSafe [22] suggests a machine learning based defense mechanism that needs to be retrained every time the environment changes. However, these defenses all have limitations: they either add a burden to the users (intrusive in terms of user experience) or need long training time. Before we design our own defense, to understand how users' interact with smart speakers, we conduct an online survey (see Section 3). From the survey, we understand how users use smart speakers and also learn their expectations on the potential defenses for smart speakers.

Speaker-Sonar. In this paper, we propose Speaker-Sonar, a new real-time, non-intrusive, liveness detection system for smart speakers. The Speaker-Sonar particularly focuses on defending against remote attackers which can exploit vulnerable network-connected speakers and send malicious voice commands. We carefully designed the Speaker-Sonar to meet the following goals: 1) provide non-intrusive user experience, 2) conduct real-time defense which does not require additional training time, 3) use hardware similar to the current smart home devices.

The key idea of our approach is to make sure the voice command is indeed coming from the user. In other words, we check both the direction of the voice command and the user, and then verify whether they are consistent. For this purpose, we transmit an inaudible ultrasonic sound through a smart speaker and track the user's direction by detecting her movement using doppler shift and compare it with the direction of a voice command received. Our defense provides non-intrusive user experience, since it works automatically by checking the directions of the user and the voice command without requiring additional user interventions. Furthermore, Speaker-Sonar works in real-time and does not need to be trained even when the environment is changing. Moreover, our approach can be generalized to other smart speakers as the Speaker-Sonar does not require any special equipment for transmitting and receiving signals (e.g., RF, microwave, Wifi). Actually, our system is simply built based on a raspberry pi 3b, a circular microphone array, and a normal speaker for playing music, which realistically imitates a common smart speaker.

However, building such a defense mechanism on these mediocre hardware pieces not designed for such a purpose is by no means trivial. More specifically, the small size of the microphone array, which is similar to popular smart speakers is small; the close distance between each microphone makes calculating an accurate distance through triangulation infeasible. Second, the speaker we use is not made for transmitting ultrasonic sound; the transmission signal has to be carefully designed to be inaudible in a higher volume. Third, smart speakers are often placed in an indoor environment, and such a reverberant environment (e.g., walls, furniture) makes movement detection hard. Fourth, detecting slow human activities with a continuous-wave radar is challenging; missing slow movements will lead to failure in the consistency check.

We address these challenges by leveraging a series of techniques: instead of using location, we use the accurate direction and the energy of the movement. We design our own wide-band signal which is inaudible and suitable for reverberant indoor environments; accordingly, for localization, we use the SRP-PHAT-HSDA, which is lightweight and accurate. To detect slow movements, we combine appropriate windowing, spectral subtracting with noise bin removal techniques to enhance the frequency spectrum for better localization results.

In the following, we elaborate on how we defend against remote attacks using the Speaker-Sonar. As discussed, the key idea behind our approach is to do a *consistency check* on the direction of the user and the voice command. Consider that the attacker has a hacked speaker under control. In the absence of Speaker-Sonar, the attacker only needs to send out a malicious voice command through the hacked speaker to attack the smart speaker. However, in the presence of the Speaker-Sonar, which does a *consistency check*, an attack no longer works without the following information: 1) where the hacked speaker is, 2) where the smart speaker is, and 3) where the user is in real-time. However, the attacker usually does not know any of such information. In such a case, the attack becomes very opportunistic.

As shown in Section 5, Speaker-Sonar is not only non-intrusive, but also is effective. Furthermore, we show that by design the Speaker-Sonar, based on consistency check, raises the bar higher for remote attacks and can defeat them.

Contributions. The contributions of the paper are outlined as follows:

• We present Speaker-Sonar, a sonar-based defense system which introduces a simple but effective consistency check; our system provides non-intrusive user experience, makes the remote attack very opportunistic, and does not require any prior training. Also, the Speaker-Sonar is effective, capable of rejecting remote voice attacks with an average accuracy of 95.5% in 2 meters radius. Furthermore, our system, based on consistency check, is able to effectively defend against all known attack techniques for creating malicious voice commands in a remote attack, to the best of our knowledge.

• We build a sonar-based radar with mediocre devices. While techniques used in such radar systems are mature, building such radar systems with devices not made for such a purpose is challenging. Throughout the paper, we described our innovations that address the challenges: e.g., designing transmit signal, windowing, *doppler-spectrum* enhancement techniques, departure detection, etc.

• We provide new understanding by conducting an online survey and show how users interact with smart speakers and what they may expect for potential defenses. Specifically, our online survey shows that more than 80% of users of both the Amazon Echo (84.5%) and Google Home (83.1%) do not feel uncomfortable coming closer (2 meters) to the smart speakers to get protected while giving critical commands.

2 BACKGROUND

In this section, we discuss the techniques and concepts that are required to understand our defense mechanism, in order to focus on the actual details that are specific to the Speaker-Sonar in Section 4. We also provide the signal processing prerequisite (e.g., FFT, time-domain, frequency-domain) in the Appendix for people who are not familiar with this area.

Sonar. Sonar is a technique that uses sound propagation to detect objects [37]. Sonar includes two types of technologies: passive sonar, which locates target through listening for the sound from a target, and active sonar, which emits sound pulses and listens for echoes [42]. In our research, we make use of both techniques; active sonar is utilized for detecting movement of users by emitting ultrasonic sound with a speaker and receiving the reflected sound with a microphone array (i.e., matrix voice) whereas the passive sonar, similar to human ears, is used for localizing voice commands with SSL (sound source localization) techniques. We operate both active sonar and passive sonar concurrently on a raspberry pi 3b; this is possible as the human voice and ultrasonic sound are in different frequency ranges. We use the ultrasonic sound in the 18–20 kHz range, which is commonly used in previous research [36, 47, 48]; such sound is often used as it is inaudible to most adults and can be produced and recorded by common speakers and microphones.

Moving Target Detection. There are various approaches for detecting moving objects: e.g., motion detector radar, ultra-wideband radar, SAR, etc [46] [34] [68]. Among them, the doppler radar transmits a microwave signal towards the target and analyzes how the target's motion has affected the frequency of the returned (or reflected) signal; the frequency change of the signal caused by a moving sender or receiver is a well-known phenomenon called the *doppler shift*. However, such radars cannot directly analyze doppler shift from the received frequency spectrum as the doppler shifts are always submerged by background noise and stationary noise; the reflection of the transmitted signal comes from both the moving targets and stationary objects. Therefore, an important step for radars is removing or suppressing such stationary noise from the received frequency spectrum which contains doppler shift. A common approach is the moving target indication (MTI) filter; the idea of the approach is to subtract the frequency spectrum with mainly noise from the frequency spectrum with both the doppler shift and the noise. Such a subtraction technique, which suppresses the stationary noise and leaves the doppler shift, is called *spectral subtraction*. As microwave does, the sound wave also is reflected by a human; thus, it is possible to build a movement detection radar based on doppler shift with a speaker and microphone as did in our research.

Time Difference of Arrival (TDOA). Sound localization, in general, includes the direction of arrival and TDOA. In far-field cases, DOA-based beamforming methods need a large number of microphones for highly accurate narrow-band sound source localization. TDOA-based methods with high sampling rates are commonly used methods for highly accurate wide-band sound source localization both in near-field and far-field cases [50]. Therefore, TDOA is applicable in our case considering the following aspects: 1) the microphone array (matrix-voice), which has eight microphones, is small, and the distance between the microphone pairs are close; the closer distance between microphones, the nearer a far-field begins. 2) wide-band signals are more suitable for reverberant indoor environments. The TDOA method consists of two separate steps: estimation of time-delay and location calculation. A common method of estimation of time delay is to use correlation methods, and among them, the generalized cross-correlation with Phase Transform (GCC-PHAT) is the most popular algorithm [28]. While GCC-PHAT more popular, SRP-PHAT (usually computed using GCC-PHAT with each pair of microphones [35]) and the MUSIC-based approaches (SEVD-MUSIC) are well-known for their robust performance in adverse acoustic environments. However, both approaches require a lot of computation. Recently, [35] proposed an SRP-PHAT method referred to as SRP-PHAT-HSDA for Hierarchical Search with directivity model and automatic calibration. SRP-PHAT-HSDA scans the 3D space over a coarse resolution grid and then refines search over a specific area, which makes the method convenient for low-cost embedded hardware. Therefore, in our work, we use the SRP-PHAT-HSDA as it is relatively lightweight but still robust in performance.

Table 1. Terminologies in this Paper

Term	**Meaning**
noise-spectrum	A frequency spectrum which includes stationary noise and background noise; later used for spectral subtraction when enhancing the *doppler-spectrum*; typically recorded when there is no movement.
doppler-spectrum	A frequency spectrum which includes doppler shift and *noise-spectrum*; typically recorded when there is movement.
enhanced doppler-spectrum	A frequency spectrum acquired by enhancing the *doppler-spectrum* using series of techniques.
defense radius	The maximum radius of where our defense works.

3 SURVEY

To build a usable protection for smart speakers, we launch online surveys to understand how users interact with the smart speakers. In particular, we design two surveys which target the users of major smart speakers (Amazon Alexa and Google Home). Note that the online survey is approved by IRB.

3.1 Survey Design

From the survey questions, we collect three categories of information: users' daily interactions with smart speakers, users' expectations for the defense on the smart speakers, and users' demographic information.

First, to understand users' interactions with the smart speakers, we begin with simple questions; e.g., what smart speaker they are using and how long they have been using it. We also ask the participants about their daily usage habits; e.g., the distance from the speaker when they talk to the speakers. Second, to know the users' expectations for the potential defense solutions on the smart speakers, we design questions that ask if users use sensitive voice commands and whether they are willing to make certain trade-offs for better protection. Finally, we collect users' demographic information, such as age, gender, occupation, and income.

After we design the survey questions, we run through a pilot study to improve the survey questions. Then we launch the survey on Mturk with the finalized questions. In the recruitment message, we write that we are looking for Alexa/Google Home users to study their experiences with these smart speakers. We also state that we are looking for participants that meet the following requirements: (1) 18 or above; (2) speak English fluently; (3) live in the US.

3.2 Survey Results and Analysis

From the survey, we collected a total of 411 valid responses (199 for Alexa and 212 for Google Home). In the following, we focus on understanding the users' responses while additional information (e.g., demographic, limitation, sample questions) related to the survey can be found in the Appendix.

In our survey, the participants report that they use the smart speakers from 0–20m distance; we consider 10–20m as outliers; also, for users' who report a range, we distribute it to the corresponding distance. From the result, we found that many users give commands in a close distance; 85.04% of Alexa users and 77.7% of Google Home users report that they give commands from 0–3m while 64.01% of Alexa users and 61.3% of Google Home users report that they talk to the device from 0–2m. The complete result can be found in Table 7 of the Appendix. Finally, 3 Alexa users and 18 Google users report using the device at a distance of 10–20m and we count them as outliers. Note that some users' range might distribute into different sub-ranges as we just reported. We also analyze more self-reported usage data from the survey. We find that users have an average of 1.54 devices at home (range from 1 to 4). Google Home users make an average of 5.98 voice commands daily, whereas Alexa users make an average of 7.28 voice commands.

We also analyzed users' expectations for the security of the smart speakers, and find that users are okay with tradeoffs such as location recording on a device for better security. According to our survey results, we find that 13.7% of Google survey participants and 10.6% of Alexa survey participants use sensitive voice commands daily. In particular, we find that most participants are okay with the tradeoffs for protecting themselves. Also, the participants are more willing to cope with the tradeoffs when they use sensitive voice commands. In general, 75.0% of the Google Survey participants (159 out of 212) and 74.9% of the Alexa survey participants are willing to come to 2m for giving commands, while 83.1% of the Google survey participants (176 out of 212) and 84.5% of the Alexa survey participants (168 out of 199) are okay with the 2m limitation when they give sensitive commands. 64.1% of the Google survey participants (136 out of 212) and 62.3% of the Alexa survey participants (124 out of 199) are okay with the device tracking their location for general commands. In comparison, for giving sensitive commands, 58.5% of the Google survey participants (124 out of 212) and 53.7% of the Alexa survey participants (107 out of 199) are okay with the device tracking their location.

4 SYSTEM DESIGN

The Speaker-Sonar checks whether the direction of the user and the voice command is consistent by emitting an inaudible ultrasonic sound and analyzing the reflected signal. In this section, we elaborate on how the Speaker-Sonar is designed and implemented. The last part of the section describes how we can use the Speaker-Sonar to tackle remote attacks in detail. In Table 1, we summarize the terminologies used throughout the section.

4.1 Approach Overview

Threat Model and Goal. We focus on tackling remote attacks that target smart speakers in the absence of a user by sending commands through compromised network-connected devices such as television, speakers, surveillance cameras, or baby monitors. While there can be multiple hacked speakers, we only use a single smart speaker to defend against the remote attacks; building a defense with more than one smart speakers in a distance make the problem much easier as we can localize the accurate *location* (not direction) of *users* and the *voice command*. Considering such threats, our defense system built based on the following assumptions. First, we assume that the smart speaker is at least 12 inches away from the walls (very reverberant) and are not near noisy

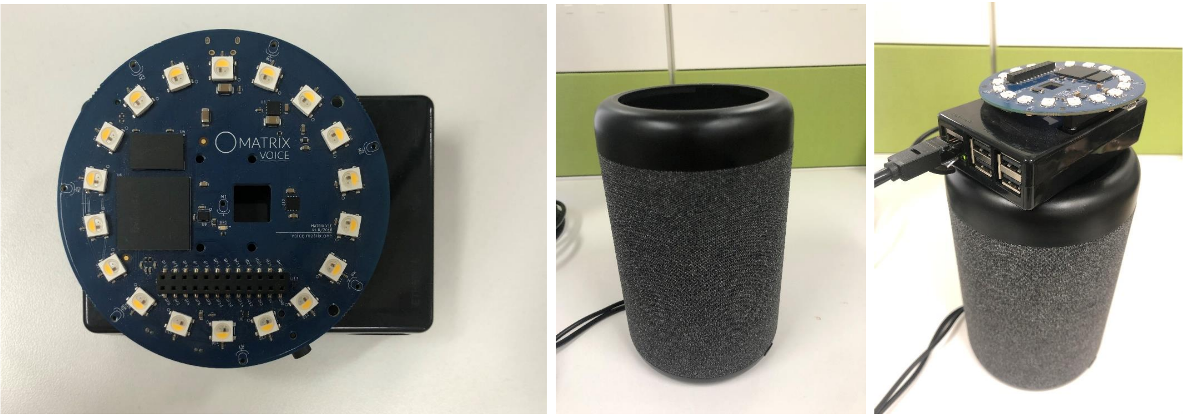

Fig. 1. Left: Raspberry pi 3b with matrix-voice. Center: Omni-directional speaker. Right: System devices together.

appliances or obstructions; common smart speakers also do not work well in such conditions and such guidelines to solve the problem can easily be found [1, 2, 6]. Second, we ask the user to remain in the same direction while saying the command; while Amazon Echo does localize the user's voice when triggered by the word *Alexa* (i.e., direction is indicated by the LED), it also does not perform active direction tracking of the user's voice command in real-time.

In this research, we aim to build a defense mechanism for smart speakers that verifies whether the direction of the user and the command the smart speaker received is consistent; in other words, we check whether the command received is indeed coming from the same direction of the user. Particularly, our mechanism focuses on delivering a non-intrusive user experience with devices that are no better than commodity smart speakers such as Amazon Echo or Google home. The prototype of our idea, Speaker-Sonar, is built using a raspberry pi 3b [18], cheap omni-directional speaker ($45 from amazon) [15] and matrix-voice ($65) [21] as shown in Figure 1. To detect and track users' direction, Speaker-Sonar emits an inaudible ultrasonic sound and analyzes the reflected signal. Building such a defense system with mediocre devices which are not built for such purpose is nontrivial.

Challenges. In the following, we list the challenges of building our approach.

- *Challenges coming from the devices.* First, the circular microphone array we are using is small (i.e., the distance between each microphone is close and ranges from 3.3cm to 7.5cm). Such a small distance makes calculating the accurate distance through triangulation infeasible as the far-field effect becomes too important; such limitation comes from antenna theory. Intuitively, if the distance between microphones is small and the sound source is in far-field, marginal errors in the direction of arrival (i.e., angle) may cause a substantial difference in distance. Because of such limitation, instead of tracking the users' location, we focus on accurately tracking the direction and use it for consistency check. Second, we use a portable speaker made for playing music. As such speakers are not made for emitting inaudible ultrasonic sound, they create sub-harmonic sounds in the audible frequency range in high volume. To solve the problem, we design our own signal with the optimal number of frequency components to make it not generate sub-harmonics at a higher volume; while planless reduction of the number of frequency components lead to inferior localization performance, too many of them make the sound audible even at a lower volume. Furthermore, to keep the defense radius at 2 meters while keeping the transmit sound inaudible, we leverage denoising and energy-based filtering techniques.
- *Challenges coming from detecting movement of users.* First, human at home usually moves relatively slow (e.g., human walking speed is around 0.7m/s - 1m/s). Such slow movements are more difficult to detect and localize

as the doppler shift made from human movement is small and overlaps with the transmit signal; the problem, reflected signals from stationary and slow-moving objects being masked by the transmit signal, is a known problem for continuous-wave radar and is challenging to solve as the limitation comes from how the approach works. We combine a series of techniques (e.g., proper windowing, noise-bin removal, spectral subtraction) and overcome such a problem. Second, while a voice command coming from the mouth can be localized to a certain direction, getting an exact direction for a user's movement is difficult; movements come from various parts of the user's body (e.g., two arms on each side of the body) which is in different direction from the microphone. Especially when the user is close to the microphone, the angle can range up to 20 to 30 degrees. Thus, for human movement, we derive a range of angles from the localized direction of the user's movements.
• *Challenges that come from the environment.* First, smart speakers, which Speaker-Sonar aims to protect, are often placed in an indoor environment. The reflected signals from stationary objects and walls in such a reverberant indoor environment make detection of human movement much more difficult. Second, the energy of the signal reflected from the human and the stationary object is very similar, as both of them are the reflection of our transmit signal. To solve both challenges, as wide-band performs better than narrow-band in a reverberant environment [54], we designed our own wide-band transmit signal, which is suitable for movement detection and use a wide-band localization algorithm (i.e., SRP-PHAT-HSDA).

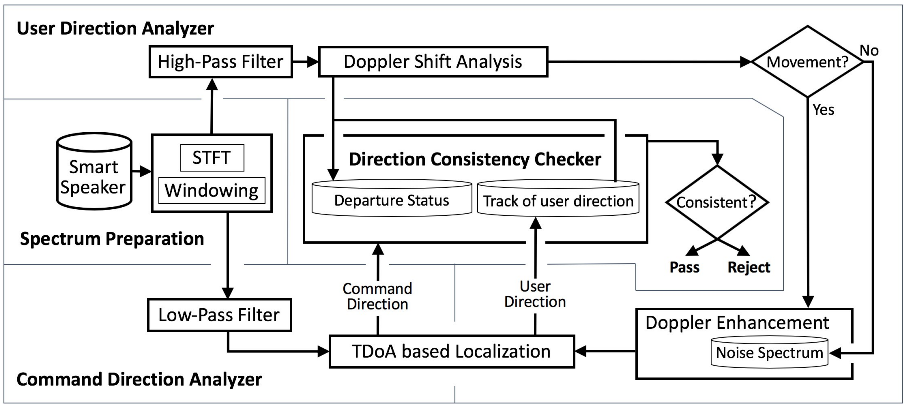

Fig. 2. Overview of Speaker-Sonar.

Design and Architecture. The design of Speaker-Sonar, illustrated in Figure 2, consists of 4 modules: *Spectrum Preparation*, *User Direction Analyzer*, *Command Direction Analyzer*, and *Direction Consistency Checker*. Their responsibilities are as follows: The *Spectrum Preparation* module prepares the frequency spectrum for the following steps by transmitting an ultrasonic sound and performing STFT (Short Time Fourier Transform) and windowing on the received signals. The *User Direction Analyzer* module gets the user's direction and detects the user's departure using a series of techniques; e.g., doppler shift analysis, energy, spectral subtraction, and TDOA (time difference of arrival) based localization algorithm. The *Command Direction Analyzer* module gets the direction of the received voice command. Finally, the *Consistency Checker* module conducts a consistency check given the direction of the user and command and the user departure status.
System Flow. Here we describe the workflow of Speaker-Sonar. The *Spectrum Preparation* module first transmits a specially designed inaudible ultrasonic sound and performs STFT (Short Time Fourier Transform) and windowing on the reflected signals received by the microphone array. The processed STFT results (i.e., frequency spectrum) are duplicated, and each of them is passed to the *User Direction Analyzer* and *Command*

Direction Analyzer module. The *User Direction Analyzer* module applies a high-pass filter to focus on the high frequencies (i.e., frequency range of the transmit signal) range and searches for doppler shifts to determine whether movement exists. When no movements are detected, the *User Direction Analyzer* module updates the noise spectrum with the input frequency spectrum. When there are movements, the input spectrum is further processed to amplify doppler shift and minimize the noise utilizing the saved noise spectrum and a series of other techniques. The *User Direction Analyzer* module then uses a TDoA based localization approach on the processed spectrum to get potential directions of the detected movements. On the other hand, once a voice command is detected, the *Command Direction Analyzer* module applies a low-pass filter to concentrate only on the human voice and similarly uses the localization approach on the input spectrum to get the direction of the command. Lastly, when both directions are obtained the *Consistency Checker* module compares the direction and decides whether to pass or reject the received command.

4.2 Spectrum Preparation

As discussed, we detect whether the user is present in an indoor area by emitting an inaudible ultrasonic sound and analyzing the reflected signal. In the following, we discuss how the transmit signal is designed and the reflected signal is processed to a frequency spectrum once recorded. To precisely detect the user's movement, the signal we emit needs to be carefully designed and processed.

Design of Transmit Sound. Indoor environments are often reverberant and have many stationary objects that reflect acoustic signals. As wide-band signals are known to work better in reverberant environments [54] we designed our own wide-band signal considering the following three points: The signal should be suitable for detecting normal indoor human activities (i.e., it needs to detect slow movements); the signal should be suitable for wide-band TDoA (time difference of arrival) localization algorithms; the signal should be inaudible but still provide enough detection range for defense. As shown in Figure 3, our transmit signal ranges from 18khz to 20.4khz and has seven peaks (i.e., frequency components) which are 400hz apart. We chose 18khz to 20.4khz as such frequency range is inaudible for most humans as other previous work does [51, 52, 55]. The signals are 400hz apart for two reasons: 1) Human walking speed approximately 1m/s [33]; for example, 1m/s speed of movement with 19khz sound can cause doppler shift of 155.7hz assuming that the speed of sound is 343m/s based on equation 4.2. Also, based on our experiment having more peaks make the sound more audible even at less volume due to sub-harmonics, which leads to a smaller detection range.

$$\Delta f = f_{ori} - f_{dop} = \frac{2v_{obj}}{v_{sound} - v_{obj}} f_{ori}.$$

Frame Size and Windowing. Processing the recorded input signal is as much important as designing the transmit signal. Speaker-Sonar uses 48khz sampling rate for input signals (i.e., recorded signals from the microphone array) and use the frame size of 4096 and hop size of 2048. Based on our experiment, 4096 provided us with enough frequency spectrum resolution for doppler shift analysis. To detect the slow movement of humans (e.g., walking at home), we designed the transmitting with 7 peaks without any other frequency components. In addition, for detecting slower movements, the type of window we use becomes important as the doppler shift can be masked (or overlap) with the stationary noise (i.e., stationary noise mostly has frequency component of the transmit signal); we cannot avoid overlap but can try to reduce the overlap as much as possible. Thus for slower movements, using a window that minimizes the side lobe but keeps the main lobe width reasonable becomes important. After testing various windows, we decided to use the blackman window over the hann window which is used in various other works [30]. While hann window can be used for detecting relatively faster movements (e.g., hand waving or repetitive motions), it turned out to be less suitable for detecting slower movements.

Table 2. *peak-bin* and *neighboring-bin* of *noise-spectrum*

Bin	1	2	3	4	5	6	7	8	9	10	11
Magnitude	0.11	0.09	0.37	1.3	7.48	14.4	12.5	4.76	0.22	0.23	0.03

Table 3. *peak-bin* and *neighboring-bin* of *doppler-spectrum*

Bin	1	2	3	4	5	6	7	8	9	10	11
Magnitude	4.11	5.52	8.37	9.3	12.48	16.2	10.5	3.76	0.22	0.24	0.02

4.3 User Direction Analyzer

The key functionality of the *User Direction Analyzer* module is to get the direction of a user and further track whether the user continues to reside within the *defense radius* (the radius Speaker-Sonar works) even when there is no movement: As shown in Figure 2, to get the direction of a movement, given the frequency spectrum from the previous module, it analyzes the doppler shifts (caused by the movement) and determines whether there is a movement or not. If there is no movement, we save the frequency spectrum as noise spectrum since the spectrum only contains background noise and stationary noise (i.e., reflected sound from stationary objects which mainly consists of frequency components of the transmit signal). The noise spectrum is later used for spectral subtraction to enhance the doppler shift. If a movement was detected, the *User Direction Analyzer* further enhances the doppler shift within the frequency spectrum using series of techniques and then utilizes a TDoA (time difference of arrival) based localization algorithm to get the direction of a user's movement. To know the presence of the user even when there is no movement, the *User Direction Analyzer* module records and keeps the last direction of the user's movement as long as the user is in the defense radius. Below we describe how each step is done in detail.

Doppler Shift Analysis. Here we elaborate on the main functionalities of Doppler Shift Analysis in Figure 2: movement detection and departure detection. To detect movements, we detect doppler shifts, a frequency change caused by movements, by analyzing the magnitude (i.e., the absolute value of any *bin*). Before we get into more details, we need to understand how the frequency spectrum with doppler shift and without it (i.e., no movement) differs from each other. When there is no movement, the frequency spectrum includes background noise and stationary noise; for short, we call such spectrum as *noise-spectrum*. When there is movement, the frequency spectrum includes *noise-spectrum* and doppler shift; for short, we call such spectrum as *doppler-spectrum*. As shown in Figure 6, the *noise-spectrum* is similar to our transmit signal Figure 3 and has no doppler shift whereas the *doppler-spectrum* has both the frequency components of the transmit signal and the doppler shift. We call the 7 *bins* that has the largest magnitude as *peak-bins*.

In the following, we further describe the details of doppler shift analysis with the examples shown in Table 2 (*noise-spectrum*) and Table 3 (*doppler-spectrum*). We begin by discussing the details of the *noise-spectrum*. As shown in Table 2, the blackman window effectively reduces the side-lobes confining the *noise-bins* (the bins that mostly contains noise) to the two *neighboring-bins* (from bin 4 to 5 and bin 7 to 8) of the *peak-bin* (bin 6). Similarly, the magnitude of the bins from 1 to 3 and 9 to 11 sharply drops to a negligible value (i.e., noise floor). Such confinement of *noise-bins* and the minimal noise floor makes the doppler shift more obvious. Table 3 further shows the details of *doppler-spectrum*. As shown, we can clearly see that the doppler shift towards the left of the *peak-bin*. Based on such observation, we consider that there exists a movement whenever the magnitude of the *non-noise-bins* (i.e., $|peakBinIndex - targetBinIndex| \geq 4$) exceed 1; with such rule we are able to consistently detect movements faster than 0.4m/s.

We elaborate on the departure detection in a separate paragraph after going through other parts as it requires user direction and its energy information.

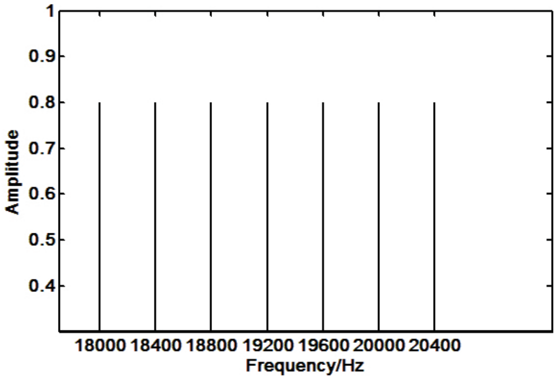

Fig. 3. Frequency Spectrum of Transmit Signal

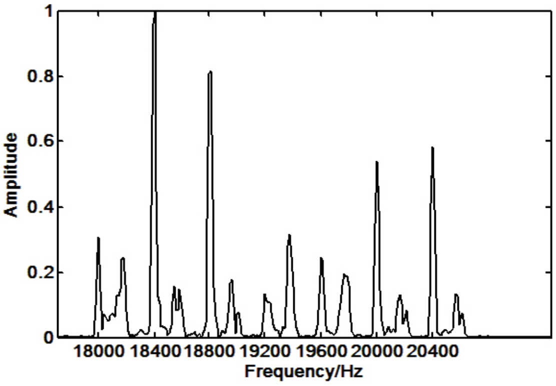

Fig. 4. Frequency Spectrum of Received Signal

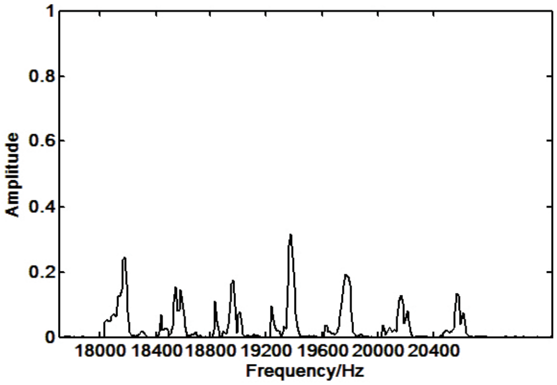

Fig. 5. Enhanced Doppler-Spectrum

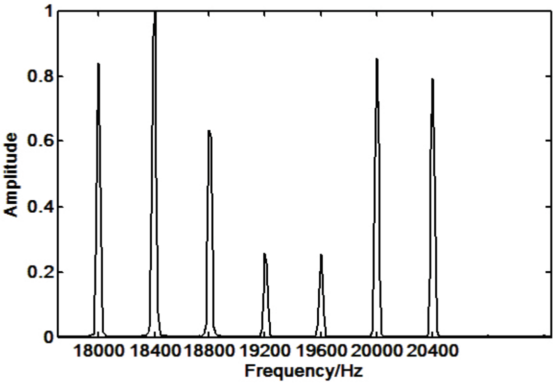

Fig. 6. Noise-Spectrum

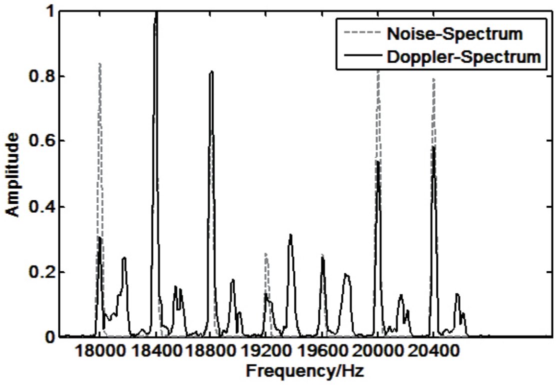

Fig. 7. Noise-Spectrum vs Doppler-Spectrum

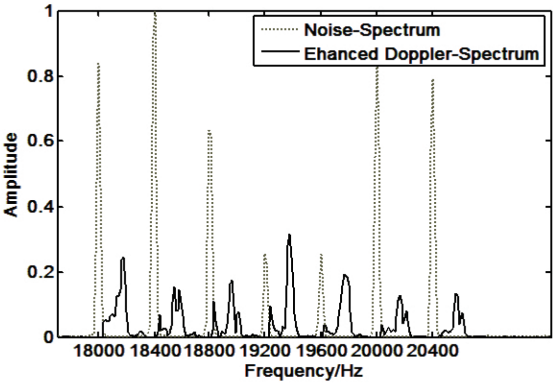

Fig. 8. Noise-Spectrum vs Enhanced Doppler-Spectrum

Doppler Enhancement. This step focuses on refining the frequency spectrum; a well-refined frequency spectrum leads to better direction detection results in the localization step. Such process is important, as sound source localization methods usually return noisy features that need to be filtered by tracking (e.g., particle filter, kalman filter) the sound sources which is expensive in terms of computing resource. However, as we are dealing with the reflected signal of our transmit signal, unlike speech which is directly received by the microphone from the sound source (i.e., has stronger energy than noise), it is not possible to differentiate movement from noise using energy. Furthermore, we avoid using such tracking algorithms as they require a lot of computation. Because of such reasons, the Doppler Enhancement step, which refines the frequency spectrum is important. To enhance the *doppler-spectrum*, we amplify the doppler shift and remove the noise using a series of techniques: magnitude spectral subtraction, *noise-bin* removal, and doppler shift amplification.

We first perform a spectral subtraction by subtracting the magnitude of the *noise spectrum* (saved in Doppler Shift Analysis step when there is no movement) from the *doppler-spectrum*. Then we add the phase of the *doppler-spectrum* to the subtraction result. All negative values were set to the noise floor before adding the phase. However, we found that such a classical but yet popular magnitude spectral subtraction approach does not work well enough for slowly moving objects as it still leaves some residue of noise. Therefore, we decided to completely remove the *noise-bins* which mainly contains the stationary noise (mostly comes from our transmit signal). Once the two steps are finished, the *doppler-spectrum* will mostly contain just the doppler shift. To make doppler shift stand out, we iterate the bins and amplify each bin with the energy of more than 0.1 Finally, the enhanced *doppler-spectrum* will further be used by the following step to get the direction of the movements.

Direction of Arrival of Movement. Once the *doppler-spectrum* is enhanced, we get the direction of the movement using SRP-PHAT-HSDA [35], an efficient state-of-art TDoA based localization algorithm. Given the enhanced *doppler-spectrum* of all microphone pairs (as we have 8 microphones, there are 28 pairs) as an input, we calculate the x, y, z coordinates (points to a direction in three-dimension) and the energy[1] of the movement utilizing SRP-PHAT-HSDA. As discussed, the potential direction of movements includes noise as completely removing noise from the frequency spectrum is inconceivable. Therefore, we further filter out the potential directions, which point to the noise utilizing the energy and their distance from other detected directions; for each frame, the direction that has the maximum Once the filtering is finished, we finally acquire the direction that corresponds to the user's movement. However, any part of the human movement can cause doppler shift and each movement would be pointed to its own direction. As a result, the detected direction of users movements can be distributed to a range of direction (i.e., angle); e.g., consider an average man with a shoulder with of 46cm approaching the smart speaker swinging his arms while walking. At the distance of 1 meter from the smart speaker, the angle human movement covers can be as wide as 20 to even 30 degrees depending on how the human is standing. To resolve this problem, we consider the movements of four frames, which covers approximately 340ms, and calculate the range of direction of the user. As we removed the potential directions which point to noise, most of the distributed potential directions from four frames are clustered pointing to a direction where the actual movements occurred. On top of the clustered potential directions, we consider the time-stamp of each direction and give more weights to recent directions and calculate the *minimum-angle* and *maximum-angle* of the movement. The *minimum-angle* and *maximum-angle* are adjusted by percentage for better performance; the detection rate based on different percentage is evaluated in Section 5. Finally, we store the direction range (i.e., *minimum-angle* and *maximum-angle*) for consistency check.

Departure Status. What we did so far is to detect the direction of the user's *live* movement in real-time. The departure status step aims to determine whether the user is still in the defense radius when there is no movement using two techniques: obvious departure detection and common behavior-based timeout. The former is for

[1] The sum of the filtered GCC-PHAT frames for all pairs of microphones provide the acoustic energy for each direction on the discrete space [35].

detecting a user promptly leaving an area, and the latter handles departure that was not detected by the former. Leveraging the two techniques, as shown in Figure 2, we update the departure status which is later used by the *Direction Consistency Checker* module with the last direction of the user for consistency check. For *non-intrusive* user experience, such a process is necessary, as we do not want to ask the user to move (to let the system know that the user is there) before saying a command. In the following, we describe the two techniques we use to update departure status in detail.

We first elaborate on how we detect obvious user departure. The departure detection is built based on the two observations: 1) when the user leaves the defense radius (area where the smart speaker is located), the direction of the doppler shift in *doppler-spectrum* will let us know that the user is getting farther from the microphone array (i.e., smart speaker); 2) the energy of the movement decreases (in the end it becomes very close to 0) as the user moves away. Firstly, to detect the direction of the doppler shift from the *doppler-spectrum*, we simply examine the *neighboring-bins* of the *peak-bin*. An increase in bandwidth on left-side of the *peak-bin* indicates that the user is moving away whereas the increase on the right-side means that the user is moving towards the smart speaker. Secondly, we use the energy variation of each direction as it reflects the distance from the movement to the smart speaker. The basic idea is to analyze the energy of the directions saved from the previous frames and check whether the energy decreases and reaches a marginal value over time. However, such trend analysis is challenging because the energy of directions fluctuates over time (i.e., not a smooth curve). Therefore, we use the Linear Least-Squares fitting algorithm to fit the energy data with a linear function. Then, using the slope of the linear function, we check the trend of the energy and see if it is increasing or decreasing. To make Speaker-Sonar efficient, we only run the departure detection whenever we detect a voice command while there is no movement; this is possible as we save the frequency spectrum, the direction and energy of user's direction of the 4 previous frames of the last movement.

Secondly, to deal with user departure missed by the obvious user departure detection approach, we use common behavior-based timeout; user departure are missed by the first technique when users leave the area slowly and non-linearly (e.g., leaving in a circular path). As we can infer from the name of the approach, the common behavior-based timeout is based on the small movements humans often make; e.g., nodding, turning the body from one direction to another, walking slowly, waving hands, standing up. While small movements with slow speed are difficult to localize (i.e., get direction), we are still able to detect the doppler shift such small movement has caused. Every time we detect such a small movement, we reset the timeout. The default timeout value is set to 5 minutes and can be adjusted based on the user's activity pattern.

4.4 Direction of Arrival of Command

Getting the direction of the command is much more simple than movement. For detecting voice commands we assume that the user says the command louder than existing background noise; such assumption is reasonable as the existing smart speakers such as the Amazon Echo and Google Home has similar requirements [1, 2, 6]. To get the Direction of Arrival of commands, we use a low-pass filter to focus on human voice and then use the SRP-PHAT-HSDA [35]. The direction of the voice can easily be obtained by selecting the direction that has the highest energy and density (i.e., many potential directions point to the same direction). Note that we keep the command direction detection part simple as the main focus of our research is to accurately detect the direction of movements and perform a consistency check utilizing a small microphone array and a commodity speaker.

4.5 Direction Consistency Checker

Once we have the direction range of the user's movement and the direction of the command, the *Direction Consistency Checker* finally decides whether to pass or reject the voice command received. If the user moved until giving the command, we simply compare the user's current direction range and the direction of the command.

However, if we receive a voice command while there is no movement, we compare it with the last location of the user's movement as long as we did not detect user departure from the defense radius. If the departure status indicates that the user is not in the defense radius, we simply reject all commands received.

4.6 Defending against Attacks

So far, we have discussed the details of how the Speaker-Sonar works. Here we discuss how we are able to tackle remote attacks with our defense system.

How the Defense Works. Whenever the *Direction Consistency Checker* decides to reject the received command, we alert the user that there was an attack attempt through an obvious feedback; when the user is not present we send an email whereas when the user is present, we give an obvious audio/visual feedback and also send an email. Moreover, the alert informs the user when and where (i.e., the direction) the malicious voice command came from and what the command was; the direction is labeled as *suspicious direction* until the user takes further action. With such information provided, the user would be able to determine whether the reported incident was an attack. Once the incident is identified as an attack, we carefully think that the user would disable the hacked device utilized by the attacker for sending malicious voice commands. In addition, we automatically block repetitive (e.g., more than two times) attack attempts coming from a particular direction; the threshold can be adjusted by the user. Thus, with the Speaker-Sonar in place, the attacker would have very few chances to attack the smart speaker.

Understanding How Attacks are Taken Down. Once again, the key idea of our defense system is to make sure that the voice command is coming from the user; in other words, with our protection in place, the attacker needs to send the malicious voice command when the *hacked speaker*, the user, and the smart speaker are in a straight line (i.e., same direction from the smart speaker). While the idea to do a consistency check is intuitive and straightforward, it makes a huge difference from the attackers perspective.

Assume that the attacker already has a network-connected device with a speaker (*hacked speaker* for short) under control. For a remote attacker to attack a smart speaker without Speaker-Sonar, the attacker can successfully attack the smart speaker by just sending the command through the *hacked speaker*. However, to attack a smart speaker with Speaker-Sonar, the attacker needs to know the following three information: 1) where the hacked speaker is, 2) where the smart speaker is, and 3) where the user is in real-time. However, the attacker usually does not know any of such information. In such a case, the success of the attack becomes very opportunistic and attacking such a system becomes virtually impossible as most likely the device would get disabled by the user before succeeding.

The Effectiveness of the Defense. Here we discuss how effective our defense is by going through each scenario. First, if the malicious command is sent in the absence of a user, most likely, all attack attempts would be detected (i.e., rejection of voice command) and reported as no user movement is detected. Second, if the malicious command is sent while the user is moving around, most likely, all attack attempts would be detected and reported; this is because we require the user to stay in a certain direction while saying the command. Third, if the malicious command is sent while the user is staying in a certain direction from the smart speaker, an attack would succeed only if the command comes from the exact same direction from the user; as mentioned, the chance for the attack to succeed is very low considering that the attacker is attacking blindly with very few chances before getting blocked. Furthermore, if the attacker used the techniques of the dolphin attack [70], which delivers voice commands through ultrasonic sound, to create a malicious voice command, the attack would have a higher chance to be blocked (i.e. by the user's body in between the *hacked speaker* and the smart speaker) even if it was sent from the same direction of the user as ultrasonic sounds cannot penetrate or go around obstacles such as human; sound that is similar to human voice have lower frequency and has a higher chance to go around (or over) human. In reality, all three scenarios should happen together throughout the day.

4.7 Limitation and Potential Threats

Limitation. The limitation of the *consistency check* is that it cannot defend against malicious voice commands sent from a hacked speaker placed in the same direction of the user. However, such limitation can only be exploited when the user is within 2 meter distance from the smart speaker. With this in mind, in the following, we discuss the potential threats.

Potential Threats. There are two attack vectors in regards to potential threats: consistency check and the user within 2 meters. We first discuss the small direction difference based voice command attack and then the voice command crafting techniques utilized in previous work, which focus on evading the user.

• *Small Direction Difference.* Such type of threat can be handled as the Speaker-Sonar can distinguish small angle differences, as shown in Section 5; the Speaker-Sonar can reject malicious commands with 90% accuracy at 20° and 80% at 10° from 1m to 2m. Since detected attack attempts get reported to the user, the attacker needs to succeed in his first try making sure that the hacked device is with the user (or very close).

• *Audible voice command.* The audible voice command, which includes attacks such as the replay attack [62, 63] and voice synthesizing attack [31, 65] can be handled with our protection in place. As we require the user to be within the 2 meter radius, the audible voice commands are likely to be heard and stopped by the user even if they pass the *consistency check*; we consider the user as an additional layer of security.

• *Inaudible voice command.* Previous research [52, 70] demonstrates inaudible voice command attacks leveraging the nonlinearity of microphones. However, both attacks require additional devices (ultrasonic transducer, amplifier, battery pack) and cannot operate directly on normal smartphones or speakers; e.g., the attacker cannot remotely hack the user's device to send such inaudible voice commands. Even if the attacker comes on-site, the attacker still needs to send the command when the user is with or near the additional device (built by the attacker) to pass the *consistency check*. Although such attacks are out of the scope of our threat model (we consider remote attacks), our protection raises the bar higher for even such sophisticated attacks.

• *Commander song.* The Commander song [69] is capable of hiding a malicious voice command in a song. Such an attack can be a threat as the attacker may play the music while the user is near the smart speaker (e.g., while using the device). However, such attacks can be mitigated using voice fingerprinting features that are embedded in popular smart speakers (e.g., Amazon Echo, Google Home).

4.8 Discussion: Additional Design and Practical Issues

Here we discuss the practical issues of Speaker-Sonar and the additional design that alleviates the issues.

Additional Design. Popular smart speakers such as the Amazon Echo have circular LED lights that are used for various functionalities; e.g., pointing the direction of the voice command it receives. Such LED lights can also be used for pointing the direction of movements of users. When users become stationary, different LED colors can be used to differentiate stationary users from actively moving users.

Practical Issues. In the following, we discuss the practical issues of the Speaker-Sonar.

• *Stationary users.* As our system relies on doppler effect, stationary users are handled by recording the last direction of users' movements (Section 4.3). The recorded last direction of the user can be indicated with the LED light before timeout. Once timeout occurs, the LED light goes off and the users would know that the recorded last direction is cleared. While sending a command, stationary users can check the LED light to see if the last direction is still recorded. If the recorded direction is cleared, users can use a common behavior (e.g., wave hand towards the speaker) that is well recognized (See Figure 9) to update the smart speaker with the current direction.

• *Multiple users.* When multiple users are present in different directions, the Speaker-Sonar recognizes the direction of the users based on their movement. If more then one users are close together (e.g., side-by-side), the system would not be able to recognize the number of users. However, such a case would not affect the *consistency check* as any commands coming from the entire direction of movement would be considered legitimate.

• *Multiple users with a stationary user.* A more complicated scenario is when a stationary user is with multiple users. When a voice command comes from an active user, the system would consider the command legitimate as it comes from a direction with movement. If the voice command comes from a stationary user with a recorded direction, the system would consider the command legitimate. However, if the recorded direction is already cleared, the stationary user would have to update the system with the current direction using a movement (e.g., wave hand).

• *Multiple users with different privilege.* The Speaker-Sonar works by detecting movements and cannot distinguish multiple users; as mentioned before, our approach aims to make sure that a command is coming from a real user (i.e., liveness detection). For differentiating users, we rely on voice fingerprinting features that are already built into popular smart speakers (e.g., Google Home, Amazon Echo).

• *Legitimate commands from more than 2 meters.* Our system works within a 2 meter radius and can cause usability issues to users who prefer to send commands from a far distance. To minimize the impact, we can perform the *consistency check* for only critical commands; for non-critical commands (e.g., turning lights on or asking weather), we can execute them, even when the user is outside of the 2 meter defense radius, and use the *consistency check* result for only monitoring purpose. Our online survey (see Section 3) also shows that more than 80% of users of both the Amazon Echo (84.5%) and Google Home (83.1%) do not feel uncomfortable coming closer (2 meters) to the smart speakers to get protected while giving critical commands.

• *Commands from non-line-of-sight location.* Our system would reject commands coming from users at non-line-of-sight locations as the system cannot detect the direction of the user's movement. Such a scenario may impact the usability of the smart speaker. Similar to the previous issue, legitimate commands from more than 2 meters, we can minimize the impact by performing the *consistency check* for only critical commands.

• *Privacy concerns.* A potential concern is the user's privacy as the Speaker-Sonar tracks the user's direction for consistency check. However, the privacy risk is minimal since the Speaker-Sonar only keeps the last direction of the user; we do not record the history of the user's direction. Moreover, the user's direction calculated within the smart speaker and never leaves it.

• *Drawbacks of using ultrasonic sounds.* Our approach is able to detect the user's direction with ordinary speakers because it uses ultrasonic sounds. However, constantly emitting ultrasonic sounds to the environment can be disturbing to pets (e.g., dogs, cats) and to people who are concerned with health-related issues.

• *Other movements in the smart-home environment.* In common households, there can be other movements coming from different sources; e.g., dog, cats, robot cleaner. However, such movements are usually closer to the floor. For such scenarios, we rely on the microphone directivity model of SRP-PHAT-HSDA [35] and filter movements coming from the floor. When the dog moves to a furniture and its movement gets detected, the attack remains opportunistic as the attacker still needs to pass the *consistency check*.

• *Tall objects.* As our approach works by detecting movements, it can differentiate between tall object and a human.

5 EVALUATION

We implemented a prototype of the Speaker-Sonar (Section 4) on top of a raspberry pi 3b, a matrix voice and an omni-directional speaker; the setup of our system is similar to the common smart speakers in the market. Our work answers the following research questions:

• RQ1: Is the consistency check performed by the Speaker-Sonar accurate?

• RQ2: Would the Speaker-Sonar work with real users?

• RQ3: Is the Speaker-Sonar effective in thwarting malicious commands sent from hacked network-connected devices?

Table 4. Accuracy of Consistency Check

			Precision (%) in Different Distances					
			0.5m	**1m**	**1.5m**	**2m**	**2.5m**	**3m**
Walk-and-Sit	**Pass**		93	97	95	90	35	20
	Reject		98	95	95	95	24	24
Walk-and-Stand	**Pass**	Pass by the Speaker	93	90	92	89	20	14
		Toward the Speaker	93	95	95	90	38	12
	Reject	Pass by the Speaker	97	97	95	92	35	18
		Toward the Speaker	98	97	95	92	40	25
Total (%)			95	95	95	91	32	19

*__Pass__: Pass rate when legitimate commands are received. *__Reject__: Rejection when malicious commands are received.

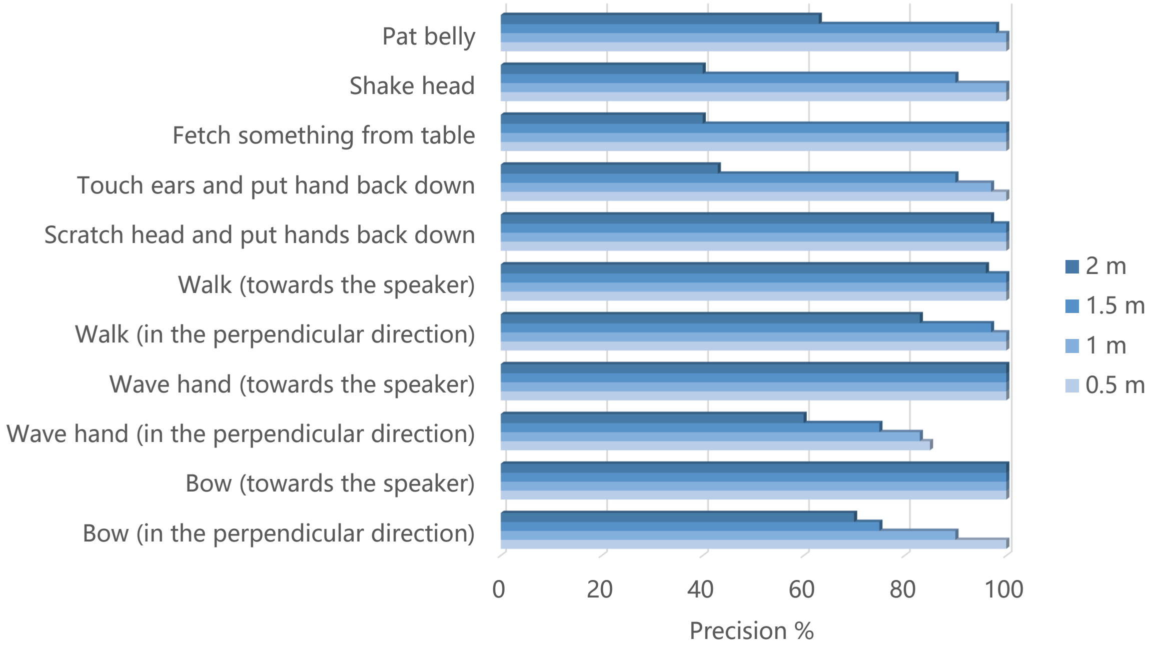

Fig. 9. Direction Detection Accuracy of Common Behaviors

The experiment was done in a real living room area with a sofa, a coffee table, two end tables, and a chair (see Figure 10). The Speaker-Sonar was placed on the coffee table without any objects blocking the speaker. All experiments were done 100 times unless otherwise mentioned.

5.1 Effectiveness of Consistency Check

To answer RQ1., we check the accuracy of the *consistency check* and further conduct a series of evaluations to provide micro-benchmarks. In our experiment, the Speaker-Sonar would *pass* the *legitimate commands* and *reject* the *malicious commands* by comparing the direction of voice command and the user.

Accuracy of Consistency Check. To measure the accuracy of the *consistency check*, we move into the defense radius and say a voice command and check whether the Speaker-Sonar is able to detect it in real-time. Within the defense radius, the user says a command following the two most common scenarios: *walk-and-stand* and *walk-and-sit*. For *walk-and-stand*, we approach the smart speaker in two ways: 1) approach the smart speaker directly facing it, 2) approach and pass the smart speaker keeping a distance. For *walk-and-sit*, we simply walk and sit on the sofa next to the smart speaker. To check how the system performs in different distances, we repeated the test from six different distances (0.5m, 1m, 1.5m, 2m, 2.5m, 3m) from the smart speaker. As shown in Table 4, the Speaker-Sonar does perform well for both the *pass* and *reject* in different distance up to 2m. An interesting result we see here is that, while *reject* success rate did not drop at 2m, *pass* success rate did drop. This is because the *pass* test needs an accurate user's direction to compare with the command's direction in order to make the correct decision. However, from 2.5m the accuracy significantly drops. This is because the signal becomes weak starting from 2m and makes the system difficult to upkeep the high accuracy. Note that the angle between the *hacked speaker* and the user was 135°.

Micro-benchmarks. In the following, we share a series of evaluation results to understand the performance of Speaker-Sonar.

• *Accuracy of Direction Detection.* To measure the accuracy of the direction detection, we perform 11 common behaviors (e.g., wave hand, pat belly, shake head) from four different distances. As shown in Figure 9, larger behaviors (e.g., wave hand and bow towards the speaker) were more precisely detected up to 2m compared to smaller behaviors (e.g., pat belly, shake head). Similarly, behaviors towards the speaker were better detected than perpendicular behaviors. Such results make sense as smaller behaviors and perpendicular behaviors create lesser a doppler effect.

• *Accuracy of Rejection in Different Angles.* To understand how effectively consistency check *rejects* malicious voice commands sent from different angles, we conducted more experiments as follows. As the *hacked speakers* can be located in various angles from the user, we conducted the rejection test on 7 different angles (i.e., 10°, 20°, 30°, 60°, 90°, 135° and 180°). As shown in Figure 14, the Speaker-Sonar rejects malicious commands sent from 20° or more with high precision (over 90%); although the accuracy drops, our system is able to reject commands coming from 10° to some extent. Such rejection accuracy of consistency check demonstrates that our defense approach indeed raises the bar for a remote attacker that *blindly* (see Section 4.6) attacks the smart speaker.

• *Impact of Reverberation.* To measure how reverberation impacts the system, we repeat the previous test, the accuracy of rejection from different angles, with the speaker placed 10cm from the wall. Surprisingly, as shown in Figure 15, the accuracy was not impacted lesser than expected; the accuracy mostly dropped when the user was at a closer distance (0.5m to 1m) from the smart speaker with small-angle difference (10° to 20°) from the hacked speaker. We carefully infer that the impact was minimal when the user was farther than 1m because the reflected sound from the wall did not contain much doppler effect; in other words, the distance was too far for the sound with doppler effect caused by the user to reach the wall and bounce back to the smart speaker. In contrast, the accuracy dropped at closer distances because the doppler shift created by the user was able to reach the smart speaker after being reflected from the wall.

• *Impact of Small Objects.* In addition to walls, interference and reverberation can also be caused by small objects. To understand the impact, we placed small objects with different sizes and texture at 0.5m distance from the smart speaker and did a *consistency check* by sending a voice command from 1m distance from the smart speaker. Note that the small objects are not blocking the line-of-sight between the user and the smart speaker. As shown in Figure 13, the impact on accuracy of *consistency check* was minimal.

• *Impact of Different Furniture Density.* To measure how different furniture density impacts the accuracy of the *consistency check*, we repeat the *walk-and-sit* evaluation with objects such as a backpack, cup, laptop, tower fan, air purifier and two plants with a vase in a room which looks like Figure 11 and Figure 12. Note that we did not place large objects that are larger than the smart speaker in between the smart speaker and the user (or hacked

Table 5. Accuracy of Consistency Check with Different Furniture Density

		Precision (%) by Range	
		0.5*m* − 1*m*	**1.5*m* − 2*m***
Walk-and-Sit	**Pass**	97	94
	Reject	97	96

***Pass**: Pass rate when legitimate commands are received. ***Reject**: Rejection when malicious commands are received.

speaker) as it is obvious that the system would not work without line-of-sight; the height of the recliner's arm is lower than the desk the smart speaker is placed. Because of the higher furniture density, we evaluate the system with a *walk-and-sit consistency check* from two locations: the recliner next to the smart speaker (0.5–1m) and from the sofa (1.5m–2m). As we see in Table 5, the results are comparable to Table 4. Such a result indicates that different furniture density does not impact the accuracy as long as there is line-of-sight.

- *Stationary Users*. As mentioned in Section 4, a stationary user is handled by recording the last direction of the user's movement. We first evaluate the system by comparing the recorded direction with the direction of the voice command sent by a stationary user, assuming that the timeout has not occurred. As expected, the result was in line with Table 4. We then cleared the recorded direction, assuming that the timeout has occurred and sent voice commands without moving, which were all rejected by the system as expected. However, as discussed in Section 4.8, stationary users can easily update their current direction with well-recognized actions such as *wave hand (towards the speaker)*.

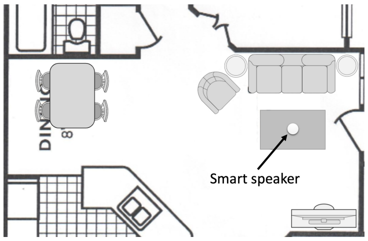

Fig. 10. Floor plan of Where Evaluation was Conducted

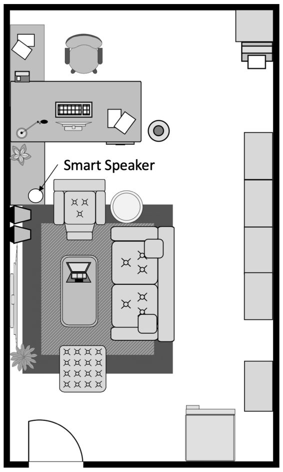

Fig. 11. Floor plan with high Furniture Density

5.2 Effectiveness with Users

To answer RQ2., with real users, we repeated the experiment summarized in Table 4 with slight modification; due to the time constraint of the user study, instead of measuring the accuracy from all four distances, we used two ranges, which are 1m to 1.5m and 1.5m to 2m. Similar to the experiment we conducted ourselves, the users were asked to follow two types of actions (*walk-and-stand* and *walk-and-sit*); we also used the same angle (135°) for rejection test. The users were allowed to approach the smart speaker from any direction of the living room. Each

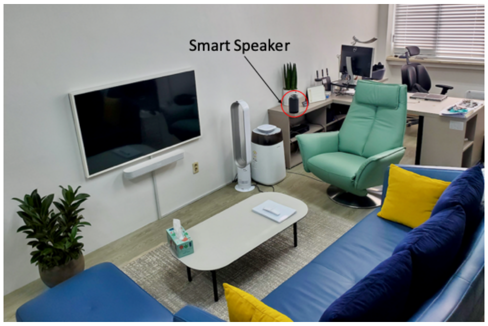

Fig. 12. Picture of Floor plan with high Furniture Density

Table 6. Accuracy of Consistency Check with Users

			Precision (%) by Range	
			$1m - 1.5m$	$1.5m - 2m$
Walk-and-Sit	**Pass**		96	92
	Reject		98	97
Walk-and-Stand	**Pass**	Pass by the Speaker	94	90
		Toward the Speaker	95	92
	Reject	Pass by the Speaker	95	94
		Toward the Speaker	96	95

*__Pass__: Pass rate when legitimate commands are received. *__Reject__: Rejection when malicious commands are received.

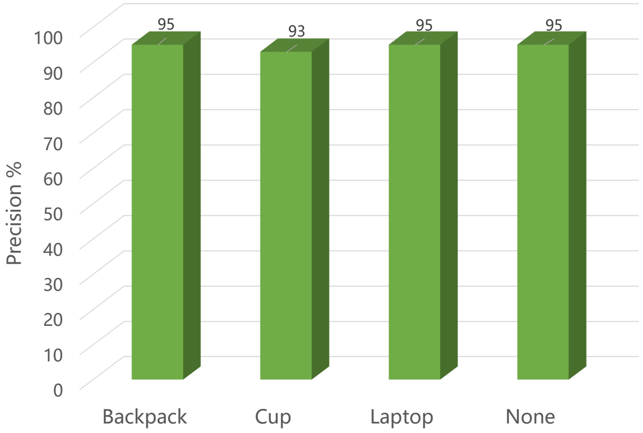

Fig. 13. Accuracy of Consistency Check with Small Objects

user study took 45–60 minutes. In total, we tested our tool with 11 users of age 18 or older in a real living room environment (see Figure 10). Also, for each user, every action was measured 10 times; in total, we collected 110 samples per action (11 users * 10 samples). As shown in Table 6, the result is comparable to the experiment we have done ourselves (see Table 4). Note that the user study was approved by the Institutional Review Board (IRB).

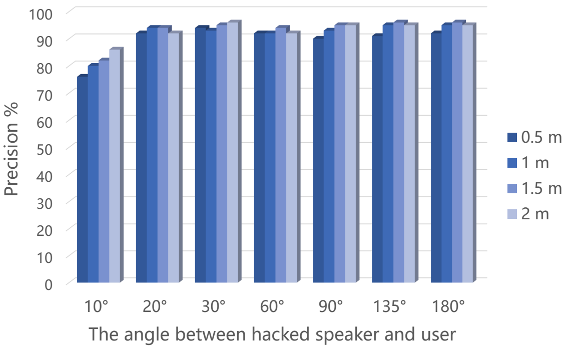

Fig. 14. Accuracy of Rejection in Different Angles

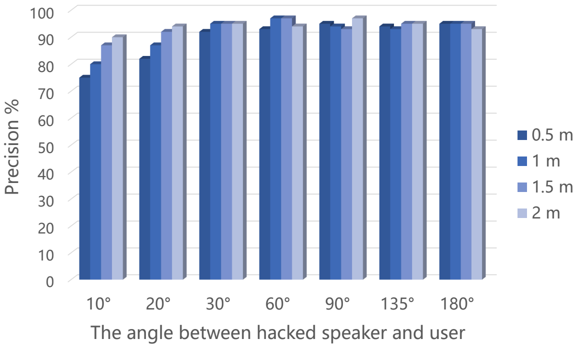

Fig. 15. Accuracy of Rejection in Different Angles with the Speaker Placed Close to the wall

5.3 Effectiveness against Remote Attacks

To answer RQ3., we evaluate our system based on three scenarios: 1) send a malicious remote command in the absence of a user, 2) send a malicious remote command in the presence of a *moving* user, 3) send a malicious remote command in the presence of a user; the user is allowed to conduct any activities such as watching TV, walking, reading books, etc.

For all three scenarios, we place four directional portable speakers, which we assume as the *hacked speakers*, in a different part of the user's room to send out malicious commands. The attacker is in another room with 4 laptops which can send commands to each of the 4 speakers in the user's room. For the first scenario, we randomly sent out malicious commands 50 times while no users were in the defense radius. As expected, by 100% we were able to reject the attack. To evaluate the second scenario, again randomly we sent out malicious commands 50 times while the user was continuously moving in the room. As expected, when the user was *continuously* moving, the rejection rate was again 100%.

For the third scenario, we add more requirements on top of the previous ones. The attacker neither knows when the user will enter the room nor leave the room. The experiment was done for an hour and the user is expected to spend 30 minutes in the room whenever he desires to. Within the hour, the attacker is allowed to send at most 120 malicious commands to the user's room. Likewise, the attacker was not able to successfully launch any attacks.

As shown in Section 5, Speaker-Sonar indeed raises the bar higher for remote attacks and shows that such attacks can effectively be defeated with the Speaker-Sonar.

5.4 Performance

To evaluate the performance of the Speaker-Sonar; we first measure the time taken for *preprocessing* (e.g., STFT, windowing); second, we analyze the time taken for calculating the direction of the user's movement and voice, given the frequency spectrum from preprocessing (*direction-analysis* for short); third, given the direction of user's movement and voice, we measure the time taken for *consistency check*; lastly, we measure the *response time* once the smart speaker is given a triggering word (e.g., "Alexa"). Response time is the total time taken which includes preprocessing, direction analysis and consistency check. For evaluation, we gave 50 voice commands to the Speaker-Sonar and measure the time taken per frame. For *preprocessing* and *direction-analysis* the average is calculated from 100 frames. The time taken for *consistency check* and *response time* is measured by getting the average from 50 frames; this is because 50 commands trigger 50 times of consistency check.

In our evaluation, *preprocessing* took 0.015 ms; *direction-analysis* took 0.11 ms; *consistency check* only took 0.0003 ms; and finally, given the triggering word (e.g., "Alexa") the response time of our system was 0.13 ms.

6 RELATED WORK

Smart speakers are becoming more ubiquitous as it is becoming the primary interaction medium between people and machine (such as smartphone, personal voice assistant, smart home appliances etc.) [10, 13, 16]. Thus, ensuring the authenticity of the voice commands leads to an active research area.

Attack on Voice Interface. Recently, a growing body of research has exploited the existing vulnerabilities that lie in voice interface [39, 41, 63, 64]. Researchers, with their sophisticated innovations, craft attacks to exploit vulnerabilities [40, 45, 60] in the voice interfaces (such as Google Assistant [16], Amazon Echo [4], Google Home [17], Apple Homepod [9] etc.). Taking control or being able to inject malicious commands into the voice interfaces enable the attackers to cause serious damage to the user (such as an unwanted purchase from the online shop [5] and malicious interactions between other smart home devices). Prior research works showed some serious attacks [29, 52, 70] which are very difficult to protect as those commands are incomprehensible to human. Recently, there happened some unintentional incidents which reveal that the smart speakers are more vulnerable to attacks [11]. Researchers demonstrated new techniques to execute a command in the smart speaker which is very easy to design. Yuan et al. demonstrated an attack by embedding adversarial voice command into a song which is recognized by the voice recognition system [69]. Besides causing serious security issues by executing malicious commands, researchers found that user privacy and sensitive information can be leaked through the smart speakers [32].

Defense on Voice Interface. While there are many pieces of research on attacks, little has been done to protect voice interfaces against those attacks. As a consequence, voice interfaces are still vulnerable to state-of-the-art attacks and can cause severe consequences to the user. Blue et al. propose to differentiate between human-generated and machine-generated voice command based on the spectrum analysis [27]. This solution needs to build a noise filter for each speaker during the initialization phase, while Speaker-Sonar can start the detection immediately. VoiceGesture [71] extracted user-specific features in the doppler shift for live user detection. On the other hand, researchers used captcha to authenticate the user when receiving a voice command [59]. However, such solutions are intrusive in terms of user experiences as they ask the user to perform additional actions. Alanwar et al. proposed EchoSafe which is a sonar-based defense mechanism against voice command attacks [22]. Whenever the room's environment changes (such as the position of furniture and other objects), Echosafe fails to perform accurately against the voice attacks. Because every time it needs to be trained for a particular orientation of the room. However, Speaker-Sonar is not affected by the change of the rooms' orientation. Furthermore, Speaker-Sonar reaches high accuracy under different scenarios. Blue et al. propose 2MA [26], which also utilizes the direction of arrival (DoA) of the voice commands to prevent remote attacks. However, 2MA requires multiple devices for localization and assumes that the user is in constant possession of their mobile device. On the other hand, our approach not only uses DoA of voice but also the movement of users and only requires a single speaker and a microphone array.

Presence detection. Researchers are able to identify the presence of people in a room with a wireless motion sensor, door sensors [44, 67]. Moreover, recent works [66] are able to compute the total number of people in a room with the help of some external hardware devices. However, Speaker-Sonar is successful in detecting human presence without the need of deploying a sensor in the room environment.

Sonar-based Localization. There is a significant amount of research [24, 25, 43] conducted using RF for localization and activity recognition. Furthermore [57] conduct sonar-based localization with ultrasonic sound utilizing special equipment (e.g., Sterling Audio ST55, Harman Kardon SoundSticks, etc.). However, not much work has been done with ultrasonic sound using commodity devices. [48] uses smart TV's and speakers to localize human over barriers but masks the transmit signal (i.e., audible) using music. [36] uses ultrasonic sound using the speaker on a laptop to infer various gestures of a moving object, [53] detects human motion with ultrasonic sound. Compare with the prior works [22], Speaker-Sonar does not require to retrain every time the environment changes.

7 CONCLUSION

In this work, we propose the Speaker-Sonar, a sonar-based defense system for smart speakers. Our defense system aims to protect the smart speakers from remote attackers that leverage network-connected speakers to send malicious commands. The key idea of our approach is to make sure that the voice command is indeed coming from the user. For such purpose, the Speaker-Sonar emits an inaudible sound and tracks the user's direction to compare it with the direction of the received voice command. The Speaker-Sonar is non-intrusive in terms of user experience as the defense works automatically doing a simple consistency check. The system we built can be generalized to smart speakers as we use similar hardware without using special equipment (e.g., RF transmitter). The Speaker-Sonar raises the bar for remote attacks and is able to effectively tackle all known attacks techniques that can be used for creating malicious voice command attacks to the best of our knowledge.

ACKNOWLEDGMENTS

We are grateful to the anonymous reviewers for their insightful comments. This work is supported in part by NSF CNS-1527141, 1618493, 1838083 and 1801432. The IIE authors are supported in part by NSFC U1836211, 61728209, National Top-notch Youth Talents Program of China, Youth Innovation Promotion Association CAS, Beijing Nova Program, Beijing Natural Science Foundation (No. JQ18011), National Frontier Science and Technology Innovation Project (No. YJKYYQ20170070).

REFERENCES

[1] [n. d.]. 8 common Amazon Echo problems and how to fix them. https://www.cnet.com/how-to/common-amazon-alexa-problems-and-how-to-fix-them/.
[2] [n. d.]. 8 common Amazon Echo problems – and how to fix them quickly. https://www.trustedreviews.com/opinion/amazon-echo-problems-2946622.
[3] [n. d.]. Alexa Skill Statistics. https://voicebot.ai/2018/03/22/amazon-alexa-skill-count-surpasses-30000-u-s/.
[4] [n. d.]. Amazon Alexa. https://developer.amazon.com/alexa.
[5] [n. d.]. Amazon Alexa ordered people dollhouses after hearing its name on TV. https://www.theverge.com/2017/1/7/14200210/amazon-alexa-tech-news-anchor-order-dollhouse.
[6] [n. d.]. Amazon Echo Not Responding Or Not Hearing You Properly? Here Are Possible Fixes. https://www.techtimes.com/articles/203474/20170329/amazon-echo-not-responding-or-not-hearing-you-properly-here-are-possible-fixes.htm.
[7] [n. d.]. Amazon Pay Is Coming To Alexa's Skills. https://www.pymnts.com/amazon/2017/payments-are-coming-to-alexas-skills/.
[8] [n. d.]. Amazon's Next Mission: Using Alexa to Help You Pay Friends. https://www.wsj.com/articles/hey-alexa-can-you-help-amazon-get-into-the-payments-business-1523007000.
[9] [n. d.]. Apple Homepod. https://www.apple.com/homepod/.
[10] [n. d.]. Apple Siri. https://www.apple.com/ios/siri/.
[11] [n. d.]. Burger King faces backlash after ad stunt backfires. https://wgntv.com/2017/04/13/burger-kings-whopper-ad-stunt/.
[12] [n. d.]. Consumer-Reports: Samsung and Roku Smart TVs Vulnerable to Hacking, Consumer Reports Finds. https://www.consumerreports.org/televisions/samsung-roku-smart-tvs-vulnerable-to-hacking-consumer-reports-finds/.
[13] [n. d.]. Cortana. https://www.microsoft.com/en-us/cortana/skills.
[14] [n. d.]. Device Vulnerabilities in the Connected Home: Uncovering Remote Code Execution and More. https://blog.trendmicro.com/trendlabs-security-intelligence/device-vulnerabilities-connected-home-remote-code-execution-and-more/.

[15] [n. d.]. GGMM D6 Portable Speaker for Amazon Echo Dot 2nd Generation, 20W Powerful True 360 Alexa Speakers. https://goo.gl/QwyMNC.

[16] [n. d.]. Google Assistant. https://assistant.google.com/.

[17] [n. d.]. Google Home. https://developers.google.com/actions/smarthome.

[18] [n. d.]. Raspberry Pi 3 Model B Motherboard in Amazon. https://www.amazon.com/Raspberry-Pi-RASPBERRYPI3-MODB-1GB-Model-Motherboard/dp/B01CD5VC92.

[19] [n. d.]. S.C. Mom Says Baby Monitor Was Hacked; Experts Say Many Devices Are Vulnerable. https://www.npr.org/sections/thetwo-way/2018/06/05/617196788/s-c-mom-says-baby-monitor-was-hacked-experts-say-many-devices-are-vulnerable.

[20] [n. d.]. Smart Speaker Users Pass 50 Million in U.S. for the First Time. https://voicebot.ai/2018/06/28/smart-speaker-users-pass-50-million-in-u-s-for-the-first-time/.

[21] [n. d.]. Voice Development Board For Everyone. https://www.matrix.one/products/voice.

[22] Amr Alanwar, Bharathan Balaji, Yuan Tian, Shuo Yang, and Mani Srivastava. 2017. EchoSafe: Sonar-based Verifiable Interaction with Intelligent Digital Agents. In *Proceedings of the 1st ACM Workshop on the Internet of Safe Things*. ACM, 38–43.

[23] Jonathan Allen. 1977. Short term spectral analysis, synthesis, and modification by discrete Fourier transform. *IEEE Transactions on Acoustics, Speech, and Signal Processing* 25, 3 (1977), 235–238.

[24] Paramvir Bahl and Venkata N Padmanabhan. 2000. RADAR: An in-building RF-based user location and tracking system. In *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, Vol. 2. IEEE, 775–784.

[25] Paramvir Bahl, Venkata N Padmanabhan, and Anand Balachandran. 2000. Enhancements to the RADAR user location and tracking system. *Microsoft Research* 2, MSR-TR-2000-12 (2000), 775–784.

[26] Logan Blue, Hadi Abdullah, Luis Vargas, and Patrick Traynor. 2018. 2MA: Verifying Voice Commands via Two Microphone Authentication. In *Proceedings of the 2018 on Asia Conference on Computer and Communications Security*. ACM, 89–100.

[27] Logan Blue, Luis Vargas, and Patrick Traynor. 2018. Hello, Is It Me You're Looking For?: Differentiating Between Human and Electronic Speakers for Voice Interface Security. In *Proceedings of the 11th ACM Conference on Security & Privacy in Wireless and Mobile Networks*. ACM, 123–133.

[28] Alessio Brutti, Maurizio Omologo, and Piergiorgio Svaizer. 2008. Comparison between different sound source localization techniques based on a real data collection. In *Hands-Free Speech Communication and Microphone Arrays, 2008. HSCMA 2008*. IEEE, 69–72.

[29] Nicholas Carlini, Pratyush Mishra, Tavish Vaidya, Yuankai Zhang, Micah Sherr, Clay Shields, David Wagner, and Wenchao Zhou. 2016. Hidden Voice Commands.. In *USENIX Security Symposium*. 513–530.

[30] Subhadeep Chakraborty. 2013. Advantages of Blackman window over Hamming window method for designing FIR filter. *International Journal of Computer Science & Engineering Technology* 4, 08 (2013).

[31] Phillip L De Leon, Michael Pucher, Junichi Yamagishi, Inma Hernaez, and Ibon Saratxaga. 2012. Evaluation of speaker verification security and detection of HMM-based synthetic speech. *IEEE Transactions on Audio, Speech, and Language Processing* 20, 8 (2012), 2280–2290.

[32] Wenrui Diao, Xiangyu Liu, Zhe Zhou, and Kehuan Zhang. 2014. Your voice assistant is mine: How to abuse speakers to steal information and control your phone. In *Proceedings of the 4th ACM Workshop on Security and Privacy in Smartphones & Mobile Devices*. ACM, 63–74.

[33] HH Dodge, NC Mattek, Daniel Austin, TL Hayes, and JA Kaye. 2012. In-home walking speeds and variability trajectories associated with mild cognitive impairment. *Neurology* 78, 24 (2012), 1946–1952.

[34] Giorgio Franceschetti, James Tatoian, David Giri, and George Gibbs. 2007. Timed arrays and their application to impulse SAR for "through-the-wall" imaging. In *Ultra-Wideband, Short-Pulse Electromagnetics 7*. Springer, 199–205.

[35] François Grondin and François Michaud. 2019. Lightweight and optimized sound source localization and tracking methods for open and closed microphone array configurations. *Robotics and Autonomous Systems* 113 (2019), 63–80.

[36] Sidhant Gupta, Daniel Morris, Shwetak Patel, and Desney Tan. 2012. Soundwave: using the doppler effect to sense gestures. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 1911–1914.

[37] Willem D Hackmann. 1986. Sonar research and naval warfare 1914-1954: A case study of a twentieth-century establishment science. *Historical Studies in the Physical and Biological Sciences* 16, 1 (1986), 83–110.

[38] Michael Heideman, Don Johnson, and C Burrus. 1984. Gauss and the history of the fast Fourier transform. *IEEE ASSP Magazine* 1, 4 (1984), 14–21.

[39] Artur Janicki, Federico Alegre, and Nicholas Evans. 2016. An assessment of automatic speaker verification vulnerabilities to replay spoofing attacks. *Security and Communication Networks* 9, 15 (2016), 3030–3044.

[40] Chaouki Kasmi and Jose Lopes Esteves. 2015. IEMI threats for information security: Remote command injection on modern smartphones. *IEEE Transactions on Electromagnetic Compatibility* 57, 6 (2015), 1752–1755.

[41] Tomi Kinnunen, Md Sahidullah, Héctor Delgado, Massimiliano Todisco, Nicholas Evans, Junichi Yamagishi, and Kong Aik Lee. 2017. The asvspoof 2017 challenge: Assessing the limits of replay spoofing attack detection. (2017).

[42] Malcolm Llewellyn-Jones. 2006. *The Royal Navy and anti-submarine warfare, 1917-49*. Routledge London.

[43] K Lorincz and M Welsh. 2004. "A Robust, Decentralized Approach to RF-Based Location Tracking,"Harvard University, Cambridge. *MA, Tech. Rep. TR-19-04, Tech. Rep.* (2004).

[44] Jiakang Lu, Tamim Sookoor, Vijay Srinivasan, Ge Gao, Brian Holben, John Stankovic, Eric Field, and Kamin Whitehouse. 2010. The smart thermostat: using occupancy sensors to save energy in homes. In *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*. ACM, 211–224.

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

[46] Soumya Nag, Mark A Barnes, Tim Payment, and Gary Holladay. 2002. Ultrawideband through-wall radar for detecting the motion of people in real time. In *Radar Sensor Technology and Data Visualization*, Vol. 4744. International Society for Optics and Photonics, 48–58.

[47] Rajalakshmi Nandakumar, Vikram Iyer, Desney Tan, and Shyamnath Gollakota. 2016. Fingerio: Using active sonar for fine-grained finger tracking. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. ACM, 1515–1525.

[48] Rajalakshmi Nandakumar, Alex Takakuwa, Tadayoshi Kohno, and Shyamnath Gollakota. 2017. Covertband: Activity information leakage using music. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 3 (2017), 87.

[49] Stefan Niewiadomski. 2013. *Filter handbook: a practical design guide.* Newnes.

[50] Ali Pourmohammad and Seyed Mohammad Ahadi. 2012. Real time high accuracy 3-D PHAT-based sound source localization using a simple 4-microphone arrangement. *IEEE Systems Journal* 6, 3 (2012), 455–468.

[51] Nirupam Roy, Haitham Hassanieh, and Romit Roy Choudhury. 2017. Backdoor: Making microphones hear inaudible sounds. In *Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services*. ACM, 2–14.

[52] Nirupam Roy, Sheng Shen, Haitham Hassanieh, and Romit Roy Choudhury. 2018. Inaudible Voice Commands: The Long-Range Attack and Defense. In *15th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 18)*. USENIX Association, 547–560.

[53] James M Sabatier and Alexander E Ekimov. 2006. *Ultrasonic methods for human motion detection.* Technical Report. MISSISSIPPI UNIV UNIVERSITY NATIONAL CENTER FOR PHYSICAL ACOUSTICS.

[54] Ville Pekka Sivonen. 2007. Directional loudness and binaural summation for wideband and reverberant sounds. *The Journal of the Acoustical Society of America* 121, 5 (2007), 2852–2861.

[55] Liwei Song and Prateek Mittal. 2017. Inaudible voice commands. *arXiv:1708.07238* (2017).

[56] Petre Stoica, Randolph L Moses, et al. 2005. Spectral analysis of signals. (2005).

[57] Stephen P Tarzia, Robert P Dick, Peter A Dinda, and Gokhan Memik. 2009. Sonar-based measurement of user presence and attention. In *Proceedings of the 11th international conference on Ubiquitous computing*. ACM, 89–92.

[58] Kazuo Toraichi, Masaru Kamada, Shuichi Itahashi, and Ryoichi Mori. 1989. Window functions represented by B-spline functions. *IEEE Transactions on Acoustics, Speech, and Signal Processing* 37, 1 (1989), 145–147.

[59] Erkam Uzun, Simon Pak Ho Chung, Irfan Essa, and Wenke Lee. 2018. rtCaptcha: A Real-Time CAPTCHA Based Liveness Detection System. (2018).

[60] Tavish Vaidya, Yuankai Zhang, Micah Sherr, and Clay Shields. 2015. Cocaine noodles: exploiting the gap between human and machine speech recognition. *WOOT* 15 (2015), 10–11.

[61] Charles Van Loan. 1992. *Computational frameworks for the fast Fourier transform.* Vol. 10. Siam.

[62] Jesús Villalba and Eduardo Lleida. 2010. Speaker verification performance degradation against spoofing and tampering attacks. In *FALA workshop*. 131–134.

[63] Zhizheng Wu, Sheng Gao, Eng Siong Cling, and Haizhou Li. 2014. A study on replay attack and anti-spoofing for text-dependent speaker verification.. In *APSIPA*. 1–5.

[64] Zhizheng Wu, Tomi Kinnunen, Nicholas Evans, Junichi Yamagishi, Cemal Hanilçi, Md Sahidullah, and Aleksandr Sizov. 2015. ASVspoof 2015: the first automatic speaker verification spoofing and countermeasures challenge. In *Sixteenth Annual Conference of the International Speech Communication Association.*

[65] Junichi Yamagishi, Takao Kobayashi, Nakano Yuji, Katsumi Ogata, and Juri Isogai. 2009. Analysis of speaker adaptation algorithms for HMM-based speech synthesis and a constrained SMAPLR adaptation algorithm. (2009).

[66] Danny B Yang, Leonidas J Guibas, et al. 2003. Counting people in crowds with a real-time network of simple image sensors. In *null*. IEEE, 122.

[67] Longqi Yang, Kevin Ting, and Mani B Srivastava. 2014. Inferring occupancy from opportunistically available sensor data. In *Pervasive Computing and Communications (PerCom), 2014 IEEE International Conference on*. IEEE, 60–68.

[68] Yunqiang Yang and Aly E Fathy. 2005. See-through-wall imaging using ultra wideband short-pulse radar system. In *Antennas and Propagation Society International Symposium, 2005 IEEE*, Vol. 3. IEEE, 334–337.

[69] Xuejing Yuan, Yuxuan Chen, Yue Zhao, Yunhui Long, Xiaokang Liu, Kai Chen, Shengzhi Zhang, Heqing Huang, Xiaofeng Wang, and Carl A Gunter. 2018. CommanderSong: A Systematic Approach for Practical Adversarial Voice Recognition. *arXiv preprint arXiv:1801.08535* (2018).

[70] Guoming Zhang, Chen Yan, Xiaoyu Ji, Tianchen Zhang, Taimin Zhang, and Wenyuan Xu. 2017. DolphinAttack: Inaudible voice commands. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 103–117.

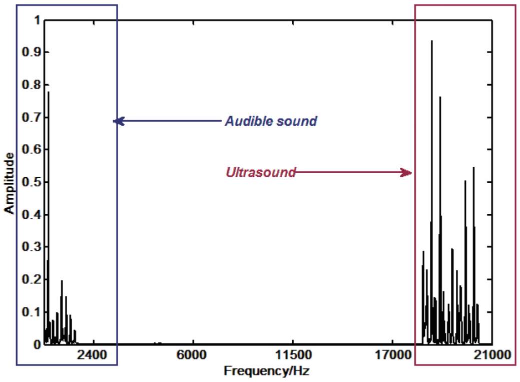

Fig. 16. Frequency spectrum of received signal

[71] Linghan Zhang, Sheng Tan, and Jie Yang. 2017. 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*. ACM, 57–71.

A APPENDIX

A.1 Signal Processing Prerequisite

Here we describe the concepts that are necessary to understand how signal (time-domain) can be processed to frequency spectrum (frequency domain). The frequency spectrum is the power spectrum of a time series input which describes the distribution of power into frequency components composing that signal [56]. Thus the spectrum provides a method to analyze a signal in terms of its frequency. For example, Figure 16 shows the frequency spectrum of a sound signal that was received by our microphone array; the left side of the figure (i.e., low frequency) is the audible sound which we use to perform sound source localization, and the right side (i.e., high frequency) is the inaudible sound we use for movement detection. The frequency spectrum of the sound wave can be obtained by Fast Fourier Transform (FFT) algorithm which samples a signal over a period of time which divides it into its frequency components [38]; the result of FFT is a single sinusoidal oscillation at distinct frequencies each with their own amplitude and phase [61]. For detecting movements (e.g., moving cars, human activities), as motion happens in a very short time, short-time Fourier transform (STFT) needs to be used which converts a short time sound signal from its time domain to frequency domain. STFT divides a longer time signal into shorter segments of equal length and then compute the FFT separately on each shorter segment [23]; the shorter segment is called *frame* and the amount of overlap is called *hop*. Revealing the spectrum on each shorter segment (*frame*) is helpful for processing movements because motions (i.e., and its spectrum) changes in a very short time. Using a shorter *frame* provides a finer and smoother movement detection result. To resolve problems such as spectral leakage which is caused by finite-length sampling, during windowing STFT a windowing function is multiplied to each *frame* [58]. Besides reducing spectral leakage, windowing function plays an important role in spectrum analysis as it helps to get the appropriate frequency spectrum for various purposes. In addition, filtering is another important concept which removes unwanted frequency components from a frequency spectrum. A high-pass filter (HPF) passes signals with a frequency higher than a certain cutoff frequency and attenuates signals with frequencies lower than the cutoff frequency [49]. Similarly, a low-pass filter (LPF) passes frequencies lower than a cutoff frequency and attenuates frequencies lower than the cutoff frequency.

Table 7. Survey result of the distance (in meter) of the user when talking to Alexa and Google Home

Distance	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
Alexa	7	57	73	45	17	5	6	-	1	1	2	1	-	-	-	-	-	-	-	-	2	-
Google	8	77	61	39	14	14	12	2	2	3	6	2	2	2	2	5	1	1	1	1	1	-

A.2 Evaluation: Supplemental Information

Departure Detection. As we discussed in Section 4, departure detection aims to detect obvious departure of a user for an immediate update on the departure status. To evaluate departure detection, we enter the defense radius and then leave after 30 seconds. The Speaker-Sonar was able to detect the user leaving the radius from less than 1m from the speaker with the accuracy of 92%; when leaving the radius from 1.5m, the accuracy dropped to 83%; leaving from 2m is not considered in our scenario as it is ambiguous to say that the user entered defense radius.

A.3 Survey: Supplemental Information

Demographic Information. The Alexa survey's participants' has diverse age level (27.6% are 26-30, 19.6% are 31-35, 16.1% are 21-25, 13.6% are 46 or older, 9.5% are 36-40, 7.5% are 41-45). 48.7% of them are female, 50.8% of them are male, 0.5% of them are others. They have diverse education levels (47.7% are bachelor, 16.6% are graduate, 16.6% are some college - no degree, 10.6% are high school graduate, 7% are associates, 1% are some high school, 0.5% are no high school). Similarly, the Google Home participants' also has diverse age level (25.9% are 26-30, 24.1% are 21-25, 17% are 31-35, 12.3% are 46 or older, 10.8% are 36-40, 5.7% are 41-45, 4.2% are 18-20). 43.9% of them are female, 54.7% of them are male, 0.5% of them are others. They have diverse education levels (48.1% are bachelor, 13.7% are graduate, 18.9% are some college - no degree, 9% are high school graduate, 9% are associates, 1.4% are some high school).

Limitations. The survey participants are recruited from Mturk, and the population distribution may not be the same as the smart speaker users. Another thing is that these results are self-reported, and might not represent the daily behaviors of the users, for example the real usage of the smart speakers, and the real reactions when they use the defenses. The results we get might be biased because of these factors. To minimize the biases, we design the questions carefully, and didn't mention things relevant to the security research we are doing, so that participants might not be leaded to think more about security than they usually do.

Sample Survey Questions

(1) What Virtual Personal Services do you use at home?
   (a) Google Home
   (b) Amazon Alexa
   (c) Both
   (d) Other
(2) Approximately now long have you used Alexa or Google Home?
   (a) 0 - 3 months
   (b) 3 - 12 months
   (c) 1 - 2 years
   (d) 2 - 3 years

   (e) 3+ years
(3) Are you in the same room when you interact with Google Home or Alexa?
   (a) Always in the same room
   (b) Always in the different room
   (c) Sometimes in the same room, and rest of the time in the other room
   (d) I don't remember
(4) How far away are you from Google Home or Alexa when you interact with it?
(5) How many Google Home or Alexa devices do you have in your home?
(6) How many voice commands do you make during a day?
(7) Do you use sensitive voice commands daily? For example, "unlock my door"
   (a) Yes
   (b) No
   (c) I'm not sure
(8) Please let us know if you are comfortable with the following scenarios. You need to come within 2 meters from the smart speaker to give a voice command.
   (a) Very uncomfortable
   (b) A little uncomfortable
   (c) Neutral
   (d) Comfortable
   (e) Very comfortable
(9) Please let us know if you are comfortable with the following scenarios. The smart speaker will track your location for better usability. For example, the smart speaker can detect if you are not around to see if the light needs to be turned off.
   (a) Very uncomfortable
   (b) A little uncomfortable
   (c) Neutral
   (d) Comfortable
   (e) Very comfortable
(10) Please let us know if you are comfortable with the following scenarios. When you make a critical voice command such as transfer money, checking bank balance etc., the smart speaker will track the location of the user to provide protection. Note that the tracking is done within the speaker and the tracked data does not leave the speaker. Also, the speaker does not save or use continuous location but only the momentary location.
   (a) Very uncomfortable
   (b) A little uncomfortable
   (c) Neutral
   (d) Comfortable
   (e) Very comfortable
(11) Please let us know if you are comfortable with the following scenarios. To be protected while making a critical command (e.g., send money to friend, open the front door), you need to come within 2 meters from the smart speaker.
   (a) Very uncomfortable
   (b) A little uncomfortable
   (c) Neutral
   (d) Comfortable
   (e) Very comfortable