CONTRIBUTED PAPER

Personal Voice Assistant Security and Privacy—A Survey

The article summarizes the state of the art in personal voice assistant security and privacy and introduces a taxonomy to structure existing research efforts.

By Peng Cheng and Utz Roedig

ABSTRACT | Personal voice assistants (PVAs) are increasingly used as interfaces to digital environments. Voice commands are used to interact with phones, smart homes, or cars. In the United States alone, the number of smart speakers, such as Amazon's Echo and Google Home, has grown by 78% to 118.5 million, and 21% of the U.S. population own at least one device. Given the increasing dependency of society on PVAs, security and privacy of these have become a major concern of users, manufacturers, and policy makers. Consequently, a steep increase in research efforts addressing security and privacy of PVAs can be observed in recent years. While some security and privacy research applicable to the PVA domain predates their recent increase in popularity, many new research strands have emerged. This article provides a survey of the state of the art in PVA security and privacy. The focus of this work is on the security and privacy challenges arising from the use of the acoustic channel. Work that describes both attacks and countermeasures is discussed. We highlight established areas such as voice authentication (VA) and new areas such as acoustic Denial of Service (DoS) that deserve more attention. This survey describes research areas where the threat is relatively well understood but where countermeasures are lacking, for example, in the area of hidden voice commands. We also discuss work that looks at privacy implications; for example, work on management of recording consent. This survey is intended to provide a comprehensive research map for PVA security and privacy.

KEYWORDS | Acoustic security and privacy; acoustic sensing; automatic speech recognition (ASR); personal voice assistant (PVA); smart speaker.

I. INTRODUCTION

The personal voice assistants (PVAs), such as the Amazon Echo, Siri, or Google Home, are now commonplace and are changing the way users interact with computer systems. Users are becoming used to interacting with devices and digitized environments, such as smart homes and cars using speech. PVAs are deployed as standalone devices, such as Amazon Echo or Google Home, and are integrated within every phone, tablet, and PC (Siri and Cortana). They are used in appliances such as TVs and set-top boxes (LG and SKYQ) and are integrated into cars (Mercedes and Jaguar). Some appliances have PVA capability without users being aware of it and others feature microphones that are dormant but can be activated by software updates [1].

Given the usefulness of PVAs, their deployment density is rapidly increasing. For example, 21% of the U.S. population own at least one smart speaker [2] and 81% of adults own a smartphone [3]. It is therefore very likely that users are always in range of at least one PVA. Users may not be aware of their presence, able to influence their behavior, or unable to deactivate them. As the devices are capable of monitoring and understanding speech, individuals have legitimate concerns regarding their privacy [4]. Users would be interested in answering questions such as: how can I control which PVAs are listening to my conversation? how can I track conversation recordings? and how can I express my privacy requirements in a

Manuscript received April 27, 2021; revised September 13, 2021 and November 13, 2021; accepted February 15, 2022. Date of publication March 11, 2022; date of current version April 1, 2022. This work was supported by the Science Foundation Ireland under Grant 19/FFP/6775. *(Corresponding author: Utz Roedig.)*

Peng Cheng is with the School of Cyber Science and Technology and the Key Laboratory of Blockchain and Cyberspace Governance of Zhejiang Province, Zhejiang University, Hangzhou 310007, China (e-mail: peng_cheng@zju.edu.cn).

Utz Roedig is with the School of Computer Science and Information Technology (CSIT), University College Cork, Cork, T12 YN60 Ireland (e-mail: u.roedig@cs.ucc.ie).

Digital Object Identifier 10.1109/JPROC.2022.3153167

world observed by numerous PVAs? Besides the obvious capability of listening to conversations, acoustic systems can also identify activities, such as laughter, crying, or eating [5]. It might also be possible to identify room characteristics such as room size or shape by analyzing recordings. Active acoustic sensing is possible too, PVA speakers can be used to emit (inaudible) sound, and reflections can be received by microphones [6]. Users would therefore like to know: how much information about my daily routines and the environment I live in is exposed? As PVAs are an interface to computer systems and smart environments, it is necessary to police access. Commands can be injected [7] without a user's ability to detect this, for example, by transmitting inaudible commands [8] or by exploiting psychoacoustic characteristics [9]. Speaker recognition can be employed to authenticate user interaction; a PVA can be trained to recognize an individual [10]. However, it is possible to circumvent identification, for example. by using recordings [11] or speech synthesis [12]. Users would therefore like to know: what possibilities exist to circumvent PVA access control? In many scenarios, resilient PVA operation is necessary. However, it is also possible to prevent a PVA from operating using Denial of Service (DoS), interfering with audio processing [13] or speech recognition (SR) [14]. Users would like to understand: what kind of DoS attack methods are possible?

This survey aims to address the aforementioned questions. We summarize the state of the art in PVA security and privacy and introduce a taxonomy to structure existing research efforts. This survey extends and enhances the taxonomy and overview of the PVA security and privacy topic outlined in the thesis by Cheng [15]. The focus of our work is on security and privacy challenges arising from the use of the acoustic channel. A PVA is a networked computer system, and as such, it is subject to general threats [16]. A PVA can be hacked and, for example, can be used as a node in a botnet. The PVA cloud infrastructure can be breached and user-specific data can be stolen. Such classical security challenges must obviously also be addressed within PVAs and associated service infrastructures. However, these are out of scope for this survey paper. Instead, our work specifically looks at issues that relate to the acoustic channel.

Work is included in this survey if it clearly falls in one of the following four categories.

1) *C1—Access Control:* The PVA's acoustic channel is used to circumvent authentication. The PVA's SR chain is manipulated to trigger unauthorized action. An example here is voice command injection.
2) *C2—Acoustic DoS:* The acoustic channel is subject to a DoS attack. The voice interface is (temporarily) disabled. A simple example here is acoustic jamming.
3) *C3—Voice Privacy:* The use of the PVA's acoustic channel leads to a loss of privacy. The recorded voice is used such that a user's privacy requirements are not met. An example here is conversation recording.
4) *C4—Acoustic Sensing:* The acoustic channel is used for an attack, not focused on the speech processing chain. The PVA's acoustic system is used to extract security-relevant information by analyzing sound. For example, the PVA is used as passive or active acoustic sonar. Although other categorization of this research domain is possible, we believe that it is useful as it aligns with perception of security and privacy in the PVA domain and existing work can be placed in these four categories.

In the next paragraph, we summarize industry and public view of security and privacy in the PVA domain. Section II gives a PVA technology background. Section III describes our security and privacy PVA research taxonomy. The following four sections describe work in the main branches of our taxonomy: C1—access control (Section IV), C2—acoustic DoS (Section V), C3—voice privacy (Section VI), and C4—acoustic sensing (Section VII). Section VIII summarizes our work and outlines areas in which we feel research advances are necessary.

A. Public PVA Security and Privacy Perception

PVAs are becoming a main interface for digital environments. The number of smart speakers in the United States has grown by 78% to 118.5 million, and 21% of the U.S. population own at least one [2]. The 2019 Australia Smart Speaker Consumer Adoption Report [17] shows that 5.7 million Australians owned smart speakers, accounting for 29.3% of the adult population. Research undertaken by Strategy Analystics [18] shows that the United Kingdom, Ireland, Canada, South Korea, Australia, Germany, and France will reach the 50% adoption threshold within the next four years.

Recently, a number of highly visible news articles have brought PVA security and privacy to the attention of the wider public. In April 2019, Amazon admitted that user recordings from PVAs are listened to by Amazon workers regularly to improve services [19]. In July 2019, Google admitted that contractors regularly listen to voice recordings obtained by their PVAs [20]. There have also been frequent reports on incidents where PVAs record or trigger action without user intent [21]. A prominent case was the interruption of U.K. MP Gavin Williamson by Apple's Siri, while he addressed the House of Commons [21]. Industry is now starting to tackle this growing public concern. For example, Amazon Alexa, Siri, and Google Home now support speaker recognition, however, mainly to distinguish speakers in a household sharing a PVA. Amazon introduced the command "Delete everything I say today" in 2019 to provide users with more privacy control. Project Alias [22] is a device that feeds a smart speaker constant white noise to disable it and provides the user with control on when to activate the PVA. Mycroft [23] is a PVA specifically designed with privacy in mind, focusing on local processing to avoid cloud analysis of recordings.

Voice data are considered sensitive data, and legal frameworks, such as the EU General Data Protection Regulation (GDPR) [24] or the California Consumer Privacy Act (CCPA) of 2018 [25], must be considered. However, with respect to recent developments of PVA technology, these existing regulations are often not clear enough and further clarification is required. Legislators in several countries are also investigating the legal context of PVA systems and it is debated if new laws are required and what form these should take. In Germany, the Parliament investigated the legality of PVA data collection and came to the conclusion that it is questionable how third parties and minors can be excluded from data collection to comply with laws [26]. Furthermore, it was deemed unclear how third parties may use data in the future after collection. In California, Assembly Bill 1395 is proposed, which would prohibit smart speaker operators from retaining or distributing voice recordings or transcriptions without the user's consent [27].

The general public is concerned about PVA security and privacy and industry and legislators are starting to react. However, as this survey shows, research has already identified much more sophisticated and serious security challenges than the ones currently triggering public debate. This survey will help to look ahead and to inform the debate.

II. PVA TECHNOLOGY

This section provides a definition of the term PVA followed by a functional description of the main PVA components. Attacks described in the remaining document usually target one specific point in the PVA processing chain that we describe here in detail.

A. PVA Definition and Ecosystem

A PVA is a device that is able to understand spoken commands and to carry out actions accordingly. A PVA contains hardware and software to record, process, and analyze sound and in most cases also contains speakers to provide users with acoustic feedback. Thus, PVAs are often referred to as smart speakers. The term smart speakers is fitting for purpose-built PVA, such as the Amazon Echo or Google Home. However, it is also possible to create a PVA by incorporating specialized software in existing computing platforms with sound processing capabilities, such as mobile phones, game consoles, or car navigation systems. In this case, solutions, such as Siri or Cortana, are referred to as intelligent assistants. In this work, we use the more general term PVA which encompasses smart speakers and intelligent assistants.

The PVA is often a distributed system and part of its functionality is located away from the device. For example, voice processing or command interpretation is usually carried out in back-end infrastructure. Command actuation is usually carried out by other Internet of Things (IoT) components.

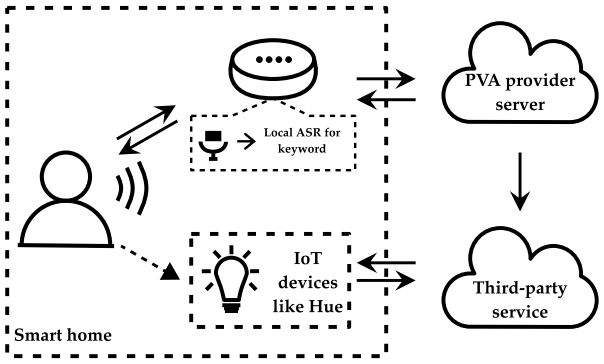


Fig. 1. *Example of how user interact with the PVA ecosystem to turn on a smart bulb (inspired by [28]).*

A PVA records sound continuously to perform wake word detection ("Alexa" in case of Amazon's Echo). Once a wake word is detected, the PVA submits the recent audio recording to a cloud-based back-end system where sophisticated Automatic Speech Recognition (ASR) is carried out. The speech is analyzed, any commands requested are executed, and a response might be formed and sent to the PVA to be played out via device speakers. Recordings are often stored in the back end and can be used for continuous ASR algorithm improvements and other services.

The aforementioned operation mode is most common, and however, variations are possible. For example, ASR might also be executed locally without involving a back end; speech may not be stored in a back end; not all PVA provide audio feedback in response to a command.

A typical PVA smart home use case is shown in Fig. 1. The user speaks a command such as Alexa, turn the light on. The PVA recognizes the keyword Alexa and the following audio Turn the light on is transported to a back end. There, ASR is used to transcribe the audio signal into text. Thereafter, the back end translates the textual command into an application programming interface (API) call to the lighting system installed in the users' home. The known location of the PVA can be used to determine the correct light to be switched ON. The back end may generate audio feedback (e.g., light has been switched ON) and send this back to the PVA to play via speaker; however, in this case, visual feedback from the light may not require additional voice feedback.

B. Audio Processing

A PVA includes at least one microphone and associated digitization processing chain, as shown in Fig. 2. It is important to consider this analog part of the processing chain as some PVA attacks focus on it.

The system usually consists of components, including a microphone, a preamplifier, a low-pass filter, and an analog-to-digital converter (ADC). There are two types of microphones: electret condenser microphones (ECMs) and microelectro mechanical systems (MEMS). However,

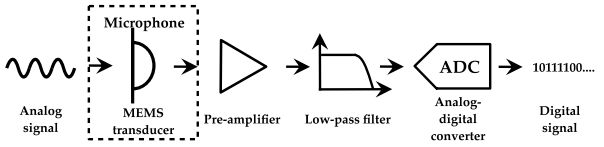


Fig. 2. *Microphone components and the sound signal flow through them [29]. Note that we do not show the sampling and hold and quantization details of the ADC but show the final digital signal instead.*

MEMS dominates the smart device market due to advantages in packaging and power consumption [8] and most commercial off-the-shelf (COTS) microphones are of this type. The microphone is a transducer converting the airborne acoustic signal into an electric signal, which only reacts to sound within the spectrum from 20 Hz to 20 kHz. The preamplifier amplifies the signal for processing in later stages. The low-pass filter removes noise above 20 kHz, which is outside the audible sound range. The ADC converts the analog signal into digital form. The sampling frequency is normally 44.1 kHz, which restricts the maximum frequency of the analog signal to 22 kHz as described by the Nyquist theorem [8].

The speaker system used in a PVA is in principle the exact reverse of the microphone. The digital signal is converted into analog form via a digital-to-analog converter (DAC). Then, the analog electrical signal is transferred to an air pressure signal via the vibration of the diaphragm of the speaker [29].

C. Automatic Speech Recognition

ASR is an interdisciplinary field of research, incorporating linguistics, computer science, and electrical engineering. The goal of SR is to transcribe speech to text automatically. A classic ASR system mimics how humans process speech. The analog acoustic signal is transformed into a digital representation, from which features are extracted. Machine learning and statistical analysis are applied to extract phonemes, the units of sound distinguishing one word from another in a language, and to finally compose text [30].

ASR is a hot topic in machine learning area and it has gone through four development stages. Gaussian mixture model–hidden Markov model (GMM-HMM) is the traditional ASR [31]; deep neural network–hidden Markov model (DNN-HMM) ASR, replacing the Gaussian mixture model (GMM) element with a deep neural network (DNN) appeared after 2012 [32]; and recurrent neural network–connectionist temporal classification (RNN-CTC) ASR replaced the hidden Markov model (HMM) with a connectionist temporal classification (CTC) [33]. In recent years, end-to-end ASR techniques using a single neural network (NN) to directly map audio input to text has drawn much attention [34]. Researchers have been studying attention/transformer-based end-to-end ASR [33]. A description of these recent developments in ASR design is out of the scope of this survey and we refer here to literature detailing this work [30].

word error rate (WER) is the most widely used performance metric for ASR evaluation. WER is defined as edit distance on a word level

$$\mathrm{WER} = (N_{\mathrm{sub}} + N_{\mathrm{ins}} + N_{\mathrm{del}})/N_{\mathrm{ref}} \tag{1}$$

where N_{sub} is the number of words that are incorrectly transcribed, N_{ins} is the number of words that appear in the current transcription but are not present in the reference, and N_{del} is the number of words in the reference that do not appear in the transcription.

Most papers detailed in this survey use WER as a basic evaluation metric. Some works also use character error rate (CER) that measures the edit distance between a generated and target transcription on a character instead of word level.

D. Attacks on ASR

The purpose of ASR is to transcribe speech to the corresponding text. An adversary is able to modify an audio signal to interfere with this process. The adversary can create a specific audio signal or modify an existing signal by adding perturbations (noise) to achieve an attack goal.

Certain signals used to attack deep learning systems are classified as obfuscated examples and adversarial examples. In the context of this survey, an obfuscated example is a signal perceived by humans as noise, while the PVA interprets a command. An adversarial example tries to fool the PVA, while it is perceived as a (benign) audio signal by humans.

We distinguish so-called targeted and nontargeted adversarial examples. In case of a targeted adversarial example, the attacker is interested in one specific command transcription, which is carefully selected. In case of a nontargeted adversarial example, the attacker does not care what specific command would be decoded by the ASR as long as the command transcription is incorrect.

Note that obfuscated examples are also called obfuscated commands in the literature. In contrast, adversarial commands are a specific type of adversarial example: a targeted adversarial example is the same as an adversarial command.

To create an either obfuscated or adversarial example, it is helpful for the attacker to have access to the internal workings of the ASR. An attack relying on internal knowledge of the ASR (e.g., the structure and parameters of the model to be attacked) is referred to as a white-box attack (see [35]–[37]). A gray-box setting means that the structure and parameters of the target model are hidden to the attacker. However, besides the final decision results, the attacker can also query the output of the last layer of the target model to acquire numerical confidence or prediction scores [38]. These values can

guide the optimization process of the attack. Finally, a black-box setting is a situation where the attacker can only get the final decision results. It is the hardest case to be considered, but it is also the most realistic scenario.

To evaluate adversarial command attacks, the attack success rate (also referred to as accuracy, effectiveness, and efficiency) is measured. The attack is successful if the ASR transcribes every single word in the target command correctly. The success rate is the percentage of successful attacks in all attack trials and measures sentence-level accuracy. The relation between success rate and WER (CER) is that a targeted adversarial example can only be treated as a successful attack when the WER and CER is 0%.

III. PVA SECURITY AND PRIVACY TAXONOMY

This survey reports on the existing security and privacy work in the PVA context with a specific focus on the acoustic channel. The survey uses the terms security and privacy in its title as both have to be considered together to cover appropriately the space that PVA users are concerned about (see Section I-A). We consider work that reports on attacks but also consider reported defense mechanisms.

Security is defined as a system condition in which system resources are free from unauthorized access and from unauthorized or accidental change, destruction, or loss (see RFC4949 [39]). Secure systems can be characterized by referring to the CIA triad [40] comprising confidentiality, integrity, and availability. Confidentiality means that data and resources are protected from unauthorized access. Integrity specifies that data and services are protected from unauthorized changes and are reliable. Availability means that data and services are available when required.

Privacy is defined as the right of a user to determine the degree to which they are willing to share their personal information with others (see RFC4949 [39]). Privacy may be compromised in three distinct ways. More information might be shared than the user has agreed to (leakage). The information that the user has agreed to share is used for purposes that the user has not consented to (purpose). The information may be stolen (breach). Security is a precondition to achieve privacy as only a secure system can prevent a breach. However, arrangements for users to consent on which data can be shared and the prevention of leakage and definition of processing purpose is beyond provision of security.

Security and privacy must also be designed with users in mind to be effective. This approach may be referred to as user-centered security, usable privacy and security, or the study of trust user experience [41]. In the PVA domain, user-focused security and privacy work is also emerging. For example, Lau *et al.* [42] investigated the mismatches between PVA controls and user needs; Bonilla and Martin-Hammond [43] studied older adults' perception of PVA privacy and transparency guidelines. This survey includes technically focused work on PVA security and privacy and we do not report work with a focus on usability.

We use the following four main categories to group existing work: C1—access control, C2—acoustic DoS, C3—voice privacy, and C4—acoustic sensing. Existing work fits well in this taxonomy and there is a clear relation between these four groups and the aforementioned view on the security and privacy domain. Categories, C1—access control and C2—acoustic DoS, are related to the term security, while categories, C3—voice privacy and C4—acoustic sensing, are related to the term privacy.

In the PVA context, a major security threat is the compromise of access control, which allows an adversary to access data and interact with services controlled by the device. Work in Category C1 is related to confidentiality in the CIA triad. The work described in C1 considers compromise of access control via the acoustic channel only and classical attacks (e.g., attacks on the OS or communication network of the PVA) are out of scope. As the compromise of access control allows an adversary to interact with PVA services and data, Category C1 of the taxonomy is also related to integrity in the CIA triad. C1 is also related to some privacy aspects as a bypass of access control is a breach, which may lead to loss of confidential data.

Category C2—DoS is related to availability in the CIA triad. In the PVA, context availability means specifically the availability of PVA services. However, in this work, we only consider DoS attacks that make use of the acoustic channel. We do not consider classical DoS attacks on other PVA system components such as the communication interface.

In the PVA, context users have privacy concerns as conversations are recorded. Users would like to keep conversations overheard by a PVA private and they would like to control how these recordings are processed. Also, speech recordings can reveal additional information about a user such as speaker identity or speaker emotions. In the context of this work, the Category C3—voice privacy is used to describe work dealing with privacy issues arising from processing of speech signals.

Category C4—acoustic sensing is also related to privacy concerns. However, different from Category C3, the acoustic channel is considered beyond the processing of voice signals. The acoustic system provided by a PVA can be used to process rich acoustic information revealing personal information for which a user may not have given consent. For example, acoustic sensing can reveal user behavior, information about the user environment, and the user itself.

Fig. 3 shows the categorization of the surveyed work according to this taxonomy. In the following paragraphs, we describe each category in more detail and explain further division in subcategories.

A. C1—Access Control

All works that fall under this category aim at circumventing access control to services a user can access via

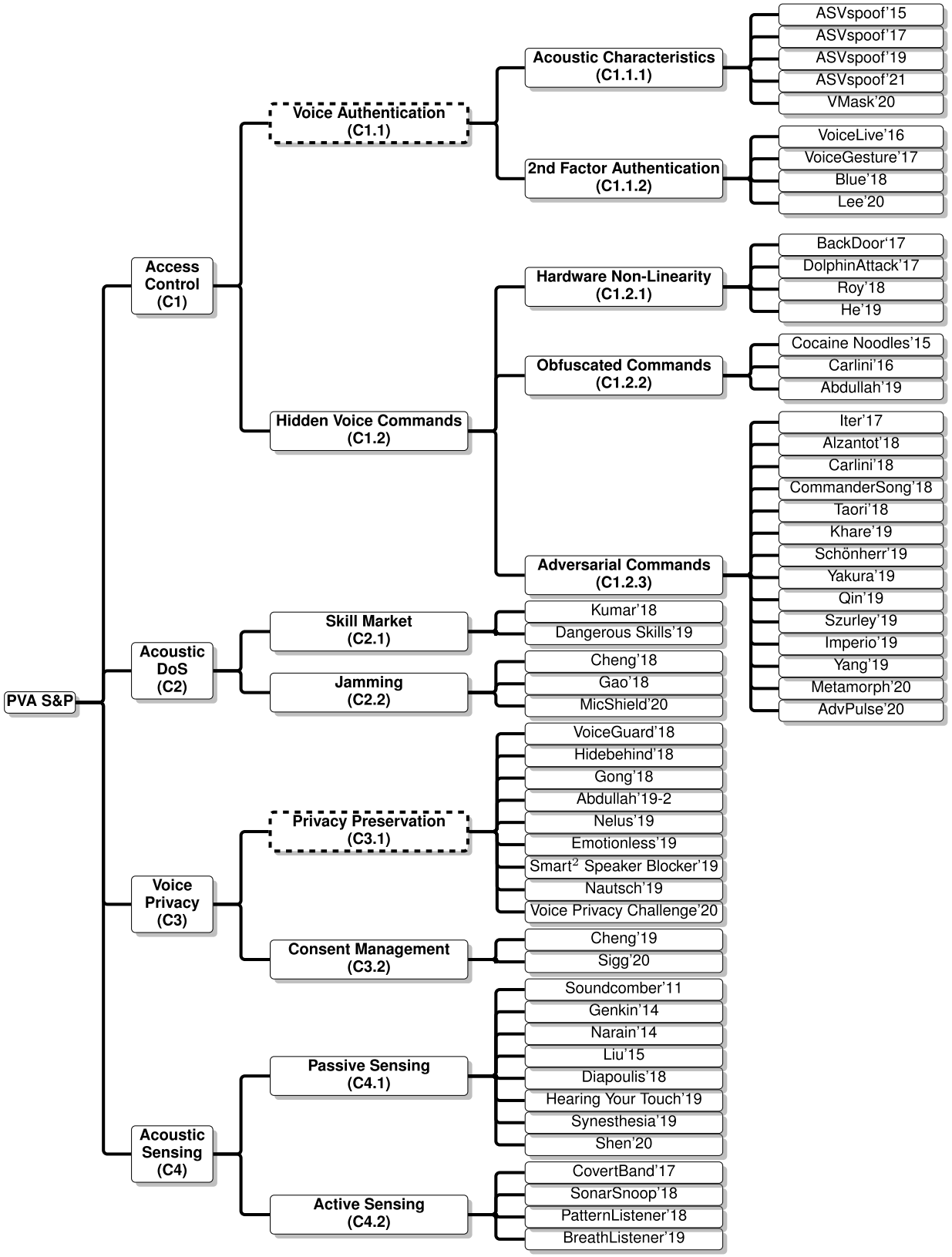


Fig. 3. *Taxonomy of PVA security and privacy challenges. Categories C1.1 and C3.1 are shown with a dashed line as these areas have significant relevance outside of the PVA domain.*

the acoustic channel of the PVA. Furthermore, work on the defense side is included, which aims to prevent bypassing the access control.

We use two subcategories here labeled voice authentication (VA) (C1.1) and hidden voice commands (C1.2).

C1.1 groups work that aims to authenticate a speaker and two significant lines of work are distinguished here: acoustic characteristics (C1.1.1) and second factor authentication (C1.1.2). C1.1.1 describes work that exploits acoustic characteristics of the speech signal for spoofing detection. C1.1.2 lists work that correlates a second source of information with the speech signal. The acoustic channel itself may provide this secondary source of information. For example, speakers and microphones can be used as a sonar system to read facial expressions of a human speaker.

It should be noted that work presented in Category C1.1 is not only relevant in the PVA context but has broader applications. However, the recent dramatic increase in the use of PVAs has highlighted the issue of VA in this context. Fig. 3 therefore shows C1.1 with a dashed box to indicate this broader context.

C1.2 contains work where the PVA's SR chain is manipulated to trigger unauthorized action. Generally, these forms of PVA attacks are referred to as command injection and the attacker aims to carry out the attack without being noticed. Three general types of hidden voice commands are distinguished: hardware nonlinearity (C1.2.1), obfuscated commands (C1.2.2), and adversarial commands (C1.2.3).

Work in Category C1.2.1 targets the analog signal processing path of a PVAs. The second class of work (C1.2.2) aims at submission of an audio signal that humans perceive as noise, while the command is understood by the PVA. Work in Category C1.2.3 aims at generating an audio signal, which is interpreted differently by humans and the ASR system.

Most works covered in this survey fall into Category C1; it is the most active research active area. However, it seems not to be the area most users are concerned with. As we outlined in Section I-A, much of the discussion in the media and under consideration by legislators in response to that are focusing on user privacy. Indeed, recent user studies such as by Abdi *et al.* [28] find that "users are mostly worried about unwanted listening from the device."

B. C2—Denial of Service

The acoustic channel can be subject to a DoS attack. The attacker aims at blocking services provided by a PVA using the acoustic channel.

Current work in this area falls in two categories: skill market (C2.1) and jamming (C2.2). Attacks in Category C2.1 aim to manipulate the back-end processing chain of the service invocation after SR. Jamming attacks (C2.2) usually target wake word recognition of the PVA.

There is little research work in this area given that PVA will be used as an interface for critical systems where continuous operation should be ensured. For example, PVAs are used to support surgeons [44] and it is vital that voice commands to operating equipment are not blocked.

C. C3—Voice Privacy

The use of the PVA's acoustic channel can lead to a loss of privacy as voice recordings may reveal sensitive user information.

Methods have been proposed to ensure that voice recordings do not reveal such user information. Methods based on hardware support, cryptographic methods, deletion, federated learning, and anonymization have been proposed.

For example, hardware support, such as Intel's Software Guard Extensions (SGX) architecture, can be leveraged. Encryption schemes, such as homomorphic encryption or secure multiparty computation, can be used to process voice data in encrypted form. Federated learning can be used to protect user data. An area of intense work is anonymization. These aim at transforming the recorded audio signal before ASR processing (potentially on a back-end server) and removing cues that may enable speaker recognition or detection of speaker characteristics such as their mood. These works are summarized under the category, privacy preservation (C3.1).

It should be noted that work presented in Category C3.1 has relevance outside of the PVA domain. A lot of works in this category are carried out without direct consideration of PVAs; however, the increasing popularity of PVAs has highlighted the importance of privacy preservation in this context. Fig. 3 therefore shows C3.1 with a dashed box to indicate this broader context.

The category consent management (C3.2) details work that aims at giving users some degree of control in regards to PVA recordings. Users would like to be able to control which PVA in their vicinity records their voice and legislators also require that users are able to provide consent.

There is substantial existing work on privacy preservation (C3.1). However, less work has investigated the issue of consent management in a PVA context (C3.2).

D. C4—Acoustic Sensing

This category summarizes work in which the acoustic channel is used for an attack, but the focus is not on the speech processing chain. The PVA's acoustic system is used to extract security and privacy relevant information by analyzing sound.

In Category C4.1, we outline existing work using passive sensing. The acoustic channel is observed and information is extracted. The attacker may use audio data recorded by the PVA itself or use an additional device. Category C4.2 describes active sensing. The PVA might also be used actively and an acoustic signal is emitted and the response is evaluated. A substantial body of work has investigated active and passive sensing in the acoustic domain, which is directly applicable to the PVA context.

It has to be noted that sensing also leads to a loss of privacy. However, the distinction between categories C3 and C4 is that while C3 purely assumes extraction of information from a voice signal, C4 is broader and considers sensing in general.

E. Paper Inclusion

In this survey, we include existing work that describes attacks and/or countermeasures in the outlined four areas: C1—access control, C2—acoustic DoS, C3—voice privacy, and C4—acoustic sensing. Sometimes, existing works in the acoustic domain do not use a PVA as an example system when describing or evaluating work. However, when it is clear that the findings can be directly applied to the PVA domain, we also include such work. For example, work on speaker recognition is generally not described in the context of PVA but is clearly a technology that could be included directly in a PVA's processing chain to facilitate authentication.

F. Naming Convention

When we describe existing work, we try to use an easily recognizable term. Many authors have given their system, attack, or countermeasure a distinct name, and in these cases, we use this established term together with the year they were published to refer to the work. Examples are DolphinAttack'17 [8] or SonarSnoop'18 [6]. If the authors have not introduced a specific name in their work, we use the combination of the lead author's surname and the year of publication as reference (e.g., Carlini'16 [45] or Cheng'19 [46]).

IV. C1—ACCESS CONTROL

In this category, we describe the work that broadly relates to the large research areas of VA and command injection. VA (Category C1.1 in this survey) is a mature research field and its results are now slowly applied to the PVA domain. It has to be noted that methods of VA are not designed as a direct countermeasure to voice injection, and however, it might be useful in this role. Injection of (hidden) voice commands (Category C1.2 in this survey) is a relatively recent research field, triggered by the appearance of PVAs.

A. Voice Authentication

VA (also known as speaker recognition or speaker authentication) is increasingly used on smart devices. Voice, among other biometric modalities such as fingerprint, facial, and iris, has been widely adopted due to two reasons. First, it can be carried out remotely over communication channels [47], and second, it represents a very natural way for users to interact with machines [48]. The goal of VA is to use voice features to identify (or verify) the identity of a speaker. Analogous to a fingerprint, the voice profile can be referred to as a voiceprint. It is solving the problem of "who is speaking," while SR addresses the issue of "what was spoken."

VA uses two steps: enrollment and authentication. During enrollment, the user is asked to provide voice samples and the unique voice features are extracted to form a user-specific model. Then, during the authentication phase, this model is used for comparison with the voice utterance to verify whether the current speaker matches the model.

Note that VA can be used in different ways. Speaker identification aims to identify to which speaker a voice belongs while considering a set of candidate voices. Speaker verification only aims to verify whether the voice belongs to the target speaker. Both variants can be implemented using similar techniques.

Some PVAs, such as Siri, already apply VA and the device is not triggered if the voiceprint does not match a legitimate profile. However, VA is not used by all PVAs. For example, the Google Assistant can still be triggered by a stranger. Smart speakers usually only check whether the keyword semantic content is correct (is "Alexa" or "Hey Google" being said?) but do not verify the voiceprint. Even though both Amazon and Google support voice recognition as reported [49], [50], these features are not used for access control but only to provide a more personalized service by linking commands to user profiles. For instance, any person or even text-to-speech (TTS) speech can trigger Amazon Echo and obtain a response [51].

Work on VA has a long tradition and started well before the relatively recent introduction of PVAs. Thus, it has to be noted that work presented here has relevance beyond the PVA context.

VA in itself is subject to attacks called spoofing attack where via replay or generated sound an attacker aims to circumvent authentication. Five different forms of spoofing attacks on VA can be distinguished: replay attacks, impersonation attacks, speech synthesis attacks, voice conversion (VC) attacks, and adversarial attacks.

A replay attack refers to a playback of a legitimate user's voice sample, prerecorded by an attacker. If the attack requires a spoken sentence that could not be prerecorded, the audio signal has to be created to match the target speaker. Impersonation attack, voice synthesis asttack (e.g., [53] and [54]), VC attack (e.g., [55] and [56]), and adversarial attack (e.g., [56]) can be used for this purpose.

Recent work on security of VA is structured around defense mechanisms against spoofing attacks. We distinguish four categories of work: acoustic characteristics, second factor authentication, copy detection, and challenge response. The first group of studies exploits acoustic characteristics of the speech signal for spoofing detection. The acoustic signal is not only analyzed for the purpose of SR and VA. In addition, signal features are used that enable the identification of a spoofed signal.

The second group of work uses a second source of information that is then correlated with the speech signal. For example, a camera can be used for lip reading and the result can be compared to the SR result. To avoid an additional sensor, the acoustic channel may also be used to

provide this secondary source of information. For example, speakers and microphones can be used as sonar systems to read articulatory gestures (VoiceGesture'17 [47]).

Copy detection aims at comparing previous voice commands to the currently issued command to ensure that each voice command is unique [57].

There is also work that employs protocols between speaker and PVA to ensure that each command is genuine. Such prompted-phrase Automatic Speaker Verification (ASV) [58], [59] has attracted much work in the past.

In the next paragraphs, we focus on work in the first two categories as these attracted most of the recent work. Simple copy detection is not seen anymore as feasible countermeasure as it is resource-intensive to keep copies of previous commands and it is too easy to circumvent this mechanism with current voice generation techniques. Challenge–response protocols can prevent attacks but are seen as impractical in a PVA context as they disturb natural speech-based user interaction.

1) Spoofing Detection via Acoustic Characteristics: Voice spoofing attacks and countermeasures have attracted a lot of interest in both research and industry communities. There is a large amount of literature, including surveys, covering this specific area [52]–[56], [60]. Thus, we do not aim to discuss exhaustively existing work in this specific area. However, we present the research progress and state of the art in this domain. ASV systems have been proved to be vulnerable to spoofing attacks. To mitigate the spoofing threat, countermeasures can be developed, which can be integrated in existing ASV.

A common problem is that it is difficult to compare the proposed spoofing countermeasures (CMs) as the different works use different datasets, experiment configurations, and evaluation metrics. To address these limitations, ASVspoof [61] has been founded. ASVspoof collects and distributes standard datasets, evaluation protocols, and metrics and facilitates competitions in which participants test their proposed algorithms. ASVspoof aims to develop generalized countermeasures effective for detecting various known and unknown attacks by controlling the prior knowledge provided to the competitors [61]. A known attack here means that the competitors are aware of the algorithm used to generate attack samples. Unknown attacks are the cases where the algorithm used to generate the attack signal is not known beforehand and it is, therefore, impossible to use this knowledge in constructing the countermeasure.

The first ASVspoof challenge was held in 2015 (referred to as ASVspoof'15 [61] in our taxonomy), and it solely focused on voice synthesis and VC attacks. Organizers provided freely accessible datasets generated by ten different synthesis and VC attack algorithms. Participants submitted their spoofing detection algorithms, which were assessed using a provided evaluation metric. The spoofing algorithms used were not accessible to the participants.

The second ASVspoof challenge was held in 2017 (referred to as ASVspoof'17 [62] in our taxonomy), and unlike the first version of the challenge, ASVspoof'17 focused on replay attack detection rather than synthesis and VC attack. The biggest challenge in detecting replay attacks from acoustic characteristics is the variations in the recorded audio samples. ASVspoof'17 tried to explore the practical limits of replay attack detection and facilitated the development of countermeasures robust enough to detect a replay attack in varying acoustic environments. ASVspoof'17 provided a baseline spoofing classifier. Subsequently, the ASVspoof organizers published the second version of the 2017 challenge dataset [63]. This dataset contains corrections for a number of anomalies, new metadata, enhancements to the original baseline system, and corresponding comparison results.

The third challenge in 2019 (referred to as ASVspoof'19 [64] in our taxonomy) introduced changes from the previous two events. First, ASVspoof'19 considered both logical access (LA) and in addition physical access (PA) scenarios. In an LA scenario, the attack signals are generated by TTS synthesis and VC techniques and are directly fed to the ASV. In a PA scenario, speech is captured by a microphone in a physical and reverberant environment (a real-world setup). This scenario is used to study replay attacks; to distinguish a voice signal from a human speaker and a replay from a loudspeaker. Second, ASVspoof'19 considers all three types of spoofing attacks (VC, synthesis, and replay attacks). Third, attack samples belonging to these types were generated using state-of-the-art neural network or waveform models, and replay attack samples resulted from a better controlled process. Finally, ASVspoof'19 took tandem decision cost function (t-DCF) as the primary new evaluation metric and used equal error rate (EER) as the secondary one. EER refers to the CM operating point where the CM miss and false acceptance rates are equivalent. t-DCF aims to assess the pooled performance of the tandem system consisting of CM and ASV.

The new metric mechanism ensured that the evaluation scores did not only appraise how well the spoofing attack detection was but also the impact of spoofing and countermeasure on the ASV system. There were 63 teams in ASVspoof'19, and more than half of them reported a spoofing detection system better than the two provided baseline systems. In the LA scenario, the best result of t-DCF of 0.0069 and EER of 0.22% is achieved by one of the teams. In the PA scenario, the best system achieves a t-DCF of 0.0096 and EER of 0.39%. For example, an EER of 0.39% means that in 0.39% of cases, the system rejects a genuine command as it falsely assumes it is spoofed. At the same time, 0.39% of spoofed commands are accepted as they are not recognized as spoofed.

The most recent ASVspoof'21 [65] has just been held. It uses more challenging data and introduces a new task involving deepfake (DF) speech detection. Results for the LA are slightly worse than results from the previous

ASVspoof edition. The best performance in the DF task is an EER of 15.64%. Finally, the results for PA show the difficulty in detecting those attacks in real and variable physical environments.

As mentioned, ASVspoof does not directly consider the PVA application context. Next, we describe the specific work of VMask'20 [38] as it represents the first practical attack targeting a PVA environment.

VMask'20 [38]: It adds perturbations to speech samples such that they are verified as belonging to a target speaker (the victim). The altered speech samples still sound as if they are spoken by the original speaker. VMask restricts the magnitude of perturbations and applies psychoacoustic masking (see also [9]). This work starts with a gray-box attack to prove the efficacy of the proposed approach and then presents a black-box attack attempt where the internal ASV structure is unknown and confidence scores cannot be obtained. This work demonstrates that ASV as a method of access control for a PVA can be circumvented.

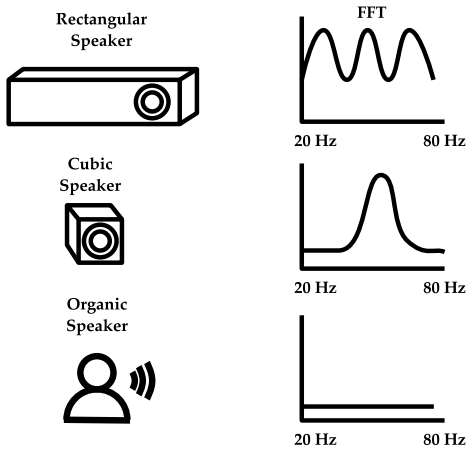


Fig. 4. *Key idea in Blue'18: audio played by COTS speakers show distinctive energy within the subbase frequency range, while there is no such phenomenon for voice of human speakers.*

2) Spoofing Detection via Second Factor: A second data source can be used to detect spoofing. In many cases, the extra information is not obtained from the acoustic channel (i.e., a camera or RF transceiver might be used). However, to avoid an additional sensor, the acoustic channel may also be used to provide this secondary source of information. Work that relies on additional data obtained from outside of the acoustic includes Chen'17 [66], Feng'17 [67], Wang'19 [68], and Pradhan'19 [69]. Works that are entirely based on data from the acoustic domain include VoiceLive'16 [70], VoiceGesture'17 [47], Blue'18 [71], and Lee'20 [72].

VoiceLive'16 [70]: It introduces a liveness detection method to defend against replay attacks targeting VA on smartphones. VoiceLive'16 is based on a user's unique vocal system and the stereo recording capabilities of a smartphone for capturing the time difference of arrival (TDoA) of phonemes. It captures TDoA changes of a sequence of phoneme sounds to the two microphones of a smartphone and calculates the TDoA similarity between the captured samples and that of the stored ones to detect whether a replay attack is in progress. This method is supposed to work as the TDoA difference between phonemes is not observed in replay attacks. The VoiceLive approach is limited to close range and a speaker's mouth must be in front of the microphone.

VoiceGesture'17 [47]: It uses the smartphone audio system (speaker and microphone) as a sonar to detect the unique articulatory gestures of a user when they speak a passphrase. VoiceGesture is a better liveliness detection system than VoiceLive'16 in regard to usability for users because VoiceGesture not only supports holding the phone in front of the mouth but also works when users hold the phone at their ears. VoiceGesture is also less susceptible to environmental noises. When a person speaks a phoneme, multidimensional articulatory movements called articulatory gestures are involved. Even if the same type of articulatory gesture is used, the movement speed and intensity may be different for different phonemes. VoiceGesture emits a 20-kHz sound wave and records all of the reflections from articulators using the built-in speaker and microphone of the phone. Analyzing the spectrogram of the recording, the movement of articulators can be revealed. Loudspeakers rely on a diaphragm that only moves in one dimension. Thus, slight differences in articulatory gestures can be used as a VA system, which is robust against replay attacks. The VoiceGesture approach is designed for the mobile phone context. However, it should be possible to use this technology in a generic PVA environment. We list this work here in Category C2.1.2—second factor authentication—as this is the main objective of this work. However, the work is also related to C4.2—active sensing as an active acoustic sensing signal is used.

Blue'18 [71]: It introduces a detection method to differentiate genuine spoken commands from replay attacks. This method explores the fact that overexcitation exists in the subbase frequency range (20–80 Hz) of audio signals generated by off-the-shelf speakers, which does not apply to a human voice (shown in Fig. 4). By identifying the rather significant energy in the subbase area, commands injected via electronic speakers can be detected. The subbase overexcitation is caused by the resonance of the enclosure material or case of the speaker. This method of spoofing detection is a useful approach for the PVA context and has less limitations regarding usability compared to VoiceLive'16 and VoiceGesture'17. However, the defense cannot work if the adversary alters the physical design of the speaker used for attack rather than just using a COTS.

Lee'20 [72]: It proposes a sonar-based framework to defend against remote attacks targeting PVAs. Remote attacks here mean that adversaries hack a network-connected smart device (such as a TV or radio) equipped

with speakers to play malicious voice commands (audible or hidden commands as introduced in Section IV-B). The core idea is to transmit an inaudible ultrasonic sound and using the Doppler shift to check the movement direction of the user while using TDoA to obtain the voice command direction. A consistency checker module compares the two directions and decides whether to accept or reject the commands. This work shows a general defense mechanism for most attacks described in Section IV. The limitations are that a hardware modification is needed and that the user needs to stay close to the PVA. Again, the work is also related to C4.2—active sensing as an active acoustic sensing signal is used.

B. Hidden Voice Commands

Work in this category investigates how to inject voice commands into a PVA without users noticing this injection, and the injected command is hidden from users in the vicinity of the PVA. The attacker aims to trigger actions without legitimate users noticing the interaction of the attacker with the PVA. In order to conceal this interaction, existing work has looked at various techniques ensuring that a person is unable to hear the submitted command, while the PVA's ASR is able to understand it. While these techniques are the essential component to enable hidden voice commands, it is also often necessary for an attacker to modify other elements of PVA interaction. After submitting a command, the PVA usually responds with a confirmation via its speakers. For example, the voice command for a home automation system "Alexa, open the front door" would result in a response "Front door opened" which an attacker would need to suppress too in order to achieve a fully hidden interaction. Research has focused on achieving hidden command injection, which we discuss here, and less work has considered how to conceal all PVA interaction.

It has to be noted that there is also work (such as by Diao *et al.* [7]), which aims to hide interaction with a PVA by carefully choosing times of command injection and volume level. Although such work can conceal interaction, these techniques do not aim to modify speech signals and are therefore not outlined in detail in this survey.

We distinguish in this survey three categories of work: C1.2.1—hardware nonlinearity, C1.2.2—obfuscated commands, and C1.2.3—adversarial commands.

Work in the first category targets the analog signal processing path of a PVA and makes use of the fact that humans are unable to hear in the high-frequency range (typically above 18 kHz). The voice command is submitted in the high-frequency space unnoticeable to users, while the nonlinear behavior of the analog signal processing path ensures that the signal is processed by the ASR.

This nonlinearity exists in the preamplifier and the microphone itself and the effect can be described similarly for both components. For example, the ideal function of an amplifier can be described as: $S_{\text{out}} = A_1 S_{\text{in}}$. The input signal S_{in} produces the linear amplified signal S_{out}. However, the nonlinearity of the amplifier will also introduce higher order signal components: $S_{\text{out}} = A_1 S_{\text{in}} + A_2 {S_{\text{in}}}^2 + A_3 {S_{\text{in}}}^3 + \cdots$ Signal components above second order can usually be neglected as they are too weak. However, the second-order component must be considered as an attacker can exploit this feature to demodulate a high-frequency inaudible attack signal to the baseband. After demodulation, this attack signal is then processed by the PVA.

Specifically, the works, BackDoor'17 [29], DolphinAttack'17 [8], Roy'18 [73], and He'19 [74], have used this approach to exploit hardware nonlinearity.

The second class of work aims at submission of an audio signal that humans perceive as noise, and the command is understood by PVAs but not by humans. For this purpose, the attacker starts with the target command, and this audio signal is gradually changed until it becomes unintelligible for a human but the PVA still decodes the command.

The purpose of ASR is to transcribe speech to corresponding text. This process can be defined as

$$y = \underset{\widetilde{y}}{\operatorname{argmax}}\, p(\widetilde{y}|x) \tag{2}$$

where x is the audio input and $\tilde{y}$ are all possible transcription candidates. The ASR aims to find the most likely transcription y given the audio input x. Once the ASR has been trained, its function is $y = f(x)$.

A human listening to the audio signal x also interprets the signal and normally would conclude that the same transcription y recognized by the ASR is the meaning of the command. This process can be described as $y = f_H(x)$ with f_H describing the human's processing capability.

An adversary can modify an input signal x by adding perturbation δ, resulting in $x' = x + \delta$. The following situation arises when an ASR decodes x'

$$y = f(x') \tag{3}$$

where y is the obfuscated command transcription, which remains the same as the one decoded from unperturbed input x. However, a human cannot perceive the same transcription y this time from the audio signal x' (it is perceived as noise; $f_H(x') = \emptyset$ with means that the human transcription is empty). The audio input x' is called the obfuscated command. We introduce later in detail Cocaine Noodles'15 [75], Carlini'16 [45], and Abdullah'19 [76] that belong to this category.

There is also another situation where $y = f_H(x')$ and $\emptyset = f(x')$. This means that the ASR is unable to transcribe the input, while a human is understanding the command well. There is work in this direction (such as work by Abdullah *et al.* [77]), which aims to prevent machines

listening into conversations. We report such work in Category C2.1—privacy preservation.

The last scenario is where $y = f_H(x')$ and $y' = f(x')$. This means that the ASR transcription and human transcription are different. This is the third class of hidden commands, the adversarial command. There have been extensive studies in the speech adversarial commands domain and we review representative works, including Iter'17 [35], Alzantot'18 [78], Carlini'18 [36], CommanderSong'18 [37], Taori'18 [79], Khare'19 [80], Schönherr'19 [9], Yakura'19 [81], Qin'19 [82], Szurley'19 [83], Imperio'19 [84], Yang'19 [85], Metamorph'20 [86], and AdvPulse'20 [87].

1) Hardware Nonlinearity: The aim is to submit a voice command in the high-frequency space unnoticeable to users, while the nonlinear behavior of the analog signal processing path ensures that the signal is processed.

BackDoor'17 [29]: has three aims: 1) to provide an acoustic but inaudible channel to a microphone; 2) to achieve high data rate inaudible acoustic communication; and 3) to provide room-level privacy protection via jamming. Although the work has a strong focus on jamming, we report the work here and not in Category C3.2—jamming as it also is the first work describing the use of the aforementioned audio nonlinearity property. The work considers frequency modulation to modulate attack signals onto a high-frequency carrier signal. For jamming, a high-frequency noise signal is used, which is demodulated due to the nonlinearity to the audible sound range and causes the ASR to fail. (The work also considers jamming that targets the Automatic Gain Control (AGC), but this method does not rely on nonlinearity effects.) The work also describes how nonlinearity features can be exploited for inaudible communication. It has to be noted that this work requires specialized transmitters and software on receiver side; thus, the work cannot be directly applied to a PVA.

The work DolphinAttack'17 [8] is similar to BackDoor'17 and advances the field in two ways. First, the messages modulated to the carrier are audio commands (not just simple signaling tones or noise). Second, DolphinAttack applies only amplitude modulation (AM) to modulate the baseband attack commands on the ultrasound carrier signal. The attack message can be demodulated to the baseband and recovered just by the nonlinearity feature of the microphones. As no additional demodulating software is needed, the attacker is capable of achieving a hidden attack on an off-the-shelf PVA. The work also investigates both hardware and software defense solutions. The former includes suppressing the ultrasound frequency sensitivity of COTS microphones and an extra module to detect the modulated attack signal for canceling the attack messages. The latter defense utilizes machine learning to classify DolphinAttack and benign audio samples based on the differences in their frequency-domain characteristics. The main limitation of this work is the requirement for very high-end equipment. Also, the attack distance is not very large (about 175 cm).

Roy'18 [73]: It builds on DolphinAttack'17 and aims at injecting commands into PVAs. DolphinAttack'17 has a limited attack range of 5 ft (175 cm, roughly 5 ft). This work introduces methods to increase the attack range to 25 ft while maintaining inaudibility of commands. Also, this work proposes a first step toward defense by proposing methods to detect nonlinearity traces. To increase the attack range, this work increases the power of speakers and addresses the accompanying audibility issue via separating parts of the AM attack signal to multiple speakers. A psychoacoustic model is also used to control the sound intensity. For defense, this work exploits the correlation between the spectrum of the sub-50-Hz band and above 50 Hz to detect the occurrence of inaudible commands injection. However, specific transmitter hardware and software are still required.

The aforementioned studies focus on the attack, while He'19 [74] focuses on defense. This work discusses limitations in the defense mechanisms proposed by DolphinAttack'17 and Roy'18 and it is shown how these methods can be bypassed. A method called active inaudible-voice-command cancellation (AIC), which detects and cancels out attack signals while retaining the legitimate command, is presented. Fig. 5 shows the defense mechanism. An attack signal is modulated to 40 kHz to attack a PVA. A guard signal is introduced, which is designed as a multitone signal with a 20-kHz interval, placing bins at W1:22 kHz, W2:42 kHz, and W3:62 kHz to frame the attack signal copy lying within 10–20 kHz (referred to as Signal 3). This will be used as a reference signal to cancel the recovered attack command (Signal 1) and Signal 2, which is an accompanying result due to the design of the guard signal. The main disadvantage of this protection approach is that this defense needs a specific speaker array for emitting the guard signal.

Comparison: A comparison between these studies on hardware nonlinearity is shown in Table 1. Modulation indicates the modulation technique used to modulate the attack commands to the high-frequency carrier signal. Demodulation describes if additional software is needed at the receiver end to recover the attack commands. Distance shows the furthest attack distance achieved. Evaluation metrics shows the main metrics used to evaluate the proposed method in the surveyed work. Defense shows if defenses against the proposed attack have been discussed. If so, are these potential defenses implemented on software or hardware level? Note that He'19 is a work that is fully focusing on defense and it is therefore not included in the table; this work requires an additional transmitter and software update on the PVA to achieve protection.

2) Obfuscated Commands: The aim of obfuscated commands is the creation of an audio signal that humans perceive as noise while the PVA interprets a command.

Table 1 Comparison of Hardware Nonlinearity Work (Category C1.2.1)

Papers	Modulation	Demodulation	Distance	Evaluation Metrics	Defence
BackDoor'17	frequency modulation	software needed	150 cm	inaudibility: user study throughput: packet error rate (PER) bit error rate (BER) jamming efficacy: word recognition rate	N/A
Dolphin-Attack'17	amplitude modulation	autonomous	175 cm	attack efficacy (recognition/activation rate)	HW/SW
Roy'18	amplitude modulation	autonomous	762 cm	attack range: distance inaudibility: sound pressure level (SPL) sound quality: ASR recognition performance defense accuracy: classification precision/recall	SW

The work Cocaine Noodles'15 [75] describes how to craft speech signals that are recognized by an ASR but are unintelligible to humans. The authors obtain different mel-frequency cepstral coefficients (MFCCs) of the original signal and convert these back to audio signals. This conversion results in mangled audio signals that are unintelligible to humans but sufficient for the ASR to transcribe.

The work Carlini'16 [45] builds on Cocaine Noodles'15 using more practical setups (i.e., greater attack distance, background noise, and newer ASR). Furthermore, the work describes a white-box attack showing that by knowing the parameters of the system, commands that are better hidden from human ears can be crafted. An attack utilizing a gradient descent (GD) approach to find optimal perturbations on the input waveform to generate the target MFCCs is used. Finally, defense mechanisms are evaluated and two defenses based on filtering and machine learning are proposed.

Abdullah'19 [76]: It proposes a methodology for generating hidden voice commands attacking multiple state-of-the-art ASR and speaker recognition systems without knowledge of the underlying systems. To make attack samples applicable for different ASRs, the work focuses on the signal processing phase almost every ASR needs. Signal features are perturbed that are important for the human auditory system but not for ASR recognition. This results in obfuscated commands that can still be transcribed correctly but are not understandable by humans. Over-the-line attacks (i.e., directly feeding the attack example to the ASR modules) and over-the-air attacks (i.e., playing the audio via loudspeakers) are tried.

Comparison: A comparison of these studies on obfuscated commands is shown in Table 2. Note that there are some common features of these works, so we do not include them in the table. These features are as follows: the obfuscated commands are all full sentences and their attacks have all been tested over the air in a room with echoes.

The first part of the table is the ASR column, including subcolumns type and model. Type indicates the essential

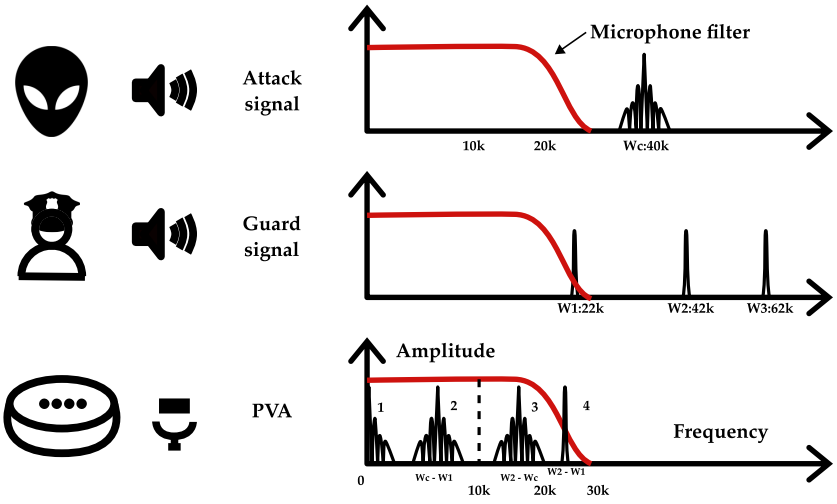


Fig. 5. ***Mix-frequent signals generated due to nonlinearity effect on the attack and the guard signal that are utilized for later attack signal cancellation in He'19.***

Table 2 Comparison ASR Obfuscated Commands Work (Category 1.2.2)

Papers	ASR		Context	Generation Tech	Perception Measurement	Evaluation Metrics
	Type	Model				
Cocaine Noodles'15	unknown	Google Now	black box	MFCC tuning	user study	human inaudibility: phoneme-level edit distance
Carlini'16	unknown & GMM-HMM	Google Now & CMU Sphinx	black box white box	MFCC tuning & GD	user study	machine comprehension: recognition rate human inaudibility: phoneme-level edit distance
Abdullah'19	multiple	multiple	black box	acoustic features tuning	psychoacoustic knowledge	only explanation & no quantifiable results

technique the ASR utilizes. Model shows the name of the targeted ASR. Abdullah'19 tests their work with various proprietary ASRs online through API and locally. Context points out whether black-box or white-box attacks are used in the study. Generation tech indicates what core techniques are used to generate the obfuscated commands and what the objective is during generation. If only experimentally adjustment is used, information is not displayed in the table. This is the case for Cocaine Noodles'15 and Abdullah'19. Perception measurement describes how the obfuscation distortion is measured. Evaluation metrics describe the metrics used in the evaluation of the proposed method. Phoneme-level edit distance is the Levenshtein edit distance between two sequences of phonemes of two transcriptions. It is used to quantifiable measure how well a human listener understands obfuscated commands. Recognition rate is the percentage of commands that are correctly interpreted.

3) Adversarial Commands: The aim is to create a signal a human perceives as an original benign command, while an ASR produces a very different transcription.

Iter'17 [35]: It is one of the earliest works in this area. The attack assumes the ASR to be a white box and targeted and nontargeted attacks are considered. Fast Gradient Sign Method (FGSM) and fooling gradient method (FGM) algorithms for producing adversarial commands are proposed. FGSM is a linear perturbation algorithm, which adds imperceptible small vectors whose elements are equal to the sign of the gradients of the cost function with respect to the input [88]. It is used to achieve an aimless adversarial attack, which results in misspellings to entire different transcriptions y' compared to the original transcription result y. Targeted adversarial commands T are generated by adding perturbations to the original input x following the guidance of the gradients of the loss function comparing target T and the temporary prediction results, which is the original prediction y gradually moving toward the target T due to the added perturbations. The perturbations are small to ensure that they are imperceptible. A gradient method is usually used to train a neural network. However, in this case, the updated objectives are no longer the parameters of the network but the inputs of the network. Hence, this method is named FGM. A user study or any metric to measure the perception distortion is missing from this study. Thus, it is not entirely understood how "hidden" the produced adversarial commands are.

Alzantot'18 [78]: It introduces a method for generating one-word targeted adversarial examples assuming a gray-box ASR. The proposed algorithm is a gradient-free genetic algorithm without knowledge of the target ASR structure and parameters. The algorithm proposed in this work takes an original audio sample x and the target command y' as inputs and then adds perturbations to x to generate a population of intermediate adversarial commands. Based on the prediction scores of the ASR for these intermediate commands, the algorithm picks candidates that fit best the target command as the base of the next generation round. The selected intermediate adversarial commands are mixed to generate a new child. The process then starts again using this child as new input. This process repeats for a predefined number of rounds or until the attack is successful; 89% of the participants in the user study cannot differentiate the adversarial command from the original.

Carlini'18 [36]: It successfully constructs white-box targeted adversarial commands for the DeepSpeech ASR [89]. A small perturbation is added to the audio input, resulting in an audio signal that is over 99.9% similar to the input. This is the first robust targeted adversarial attack study, resulting in audio output that can be influenced such that theoretically any chosen phrase can be transcribed. This work formulates the generation of the adversarial commands as an optimization problem in order to find the minimum necessary perturbations.

CommanderSong'18 [37]: It achieves an adversarial white-box attack against the Kaldi ASR by creating modified songs that are perceived by listeners as songs but recognized by the ASR as commands. Robust adversarial examples are created that can be played over the air using the gradient method but with an additional noise factor added to the song together with the perturbation. In this way, the loss optimization process can ensure that commander song can still be recognized by the ASR as

the noise in the environment has been considered during signal generation. A user study finds that none of the commands embedded in a song can be recognized by humans. Transferrability of the adversarial samples to a black-box ASR iFLYTEK is demonstrated. After using the algorithm of Carlini'18 to further modify the generated adversarial examples, they can be successfully decoded by both DeepSpeech and Kaldi. Two defense mechanisms, adding noise and downsampling, are proposed. The former lowers the signal-to-noise ratio (SNR) and is not effective for robust over-the-air examples. The latter significantly lowers the success rate of the attacks, while benign commands can still survive.

Taori'18 [79]: It introduces a new generation method for targeted adversarial commands assuming a gray-box ASR system. The work improves on Alzantot'18 as a more complex ASR system (i.e., DeepSpeech) is targeted. This work generates adversarial commands from normal samples of the Common Voice dataset such that these are interpreted as two words from the 1000 most common English words. Their algorithm combines the genetic methodology of Alzantot'18 and gradient estimation techniques.

Khare'19 [80]: It proposes a new framework using multiobjective evolutionary optimization to generate adversarial commands for both nontargeted and targeted attacks on black-box ASRs. For the generation process, a set of original audio inputs is selected and random uniform noise is added to them as the initialization. A genetic algorithm is applied to pick good genes (fitness scores are used) from candidate parent's examples to generate child examples in each iteration. The top N# candidates are selected for the next iteration. This process repeats until the fitness goal is achieved or the maximum number of iterations is reached. Compared with Taori'18, the resulting command is closer to the target phrase while maintaining comparable acoustic similarity with the original sample.

Schönherr'19 [9]: It proposes a targeted adversarial attack, tricking human perception based on the psychoacoustic model of the human auditory system. Targeting a trained white-box ASR Kaldi model, the adversarial command generation method first applies forced alignment between the original input and the target transcription to calculate the best possible temporal alignment between the original audio and the target transcription. Then, backpropagation is used to calculate the perturbations required to force the ASR to transcribe the target output. Two critical points of this work rely on this backpropagation process: 1) hearing threshold based on the original audio is calculated and applied in the backpropagation to limit the modifications, which results in changes hardly perceptible by the human auditory system, and 2) the preprocessing step is integrated with the DNN into a joint network for the backpropagation. Compared to previous targeted adversarial commands, the perturbation noise is significantly reduced as the human auditory system is considered properly. This is confirmed using a two-part audibility study consisting of a user study and a MUSHRA [90] test.

Yakura'19 [81]: It proposes a method for the generation of targeted adversarial commands against the ASR DeepSpeech. The described attack is the first over-the-air work against this type of ASR; previously described works usually feed adversarial commands directly to the ASR algorithms without taking speaker, room, and microphone characteristics into account. The method is based on a white-box assumption and incorporates transforms introduced due to playback, reverberation, and recording distortions acting on the audio signals when the attack is launched over the air. The results are compared with over-the-air results of CommanderSong'18, and the proposed method can generate samples with less perturbation while targeting a different ASR model. A user study on Amazon Turk also proves that attacks are hardly noticeable. However, as signal generation is complex and time-consuming, only a few samples are tested. This also means that the practicality of the attack is limited to situations where the attacker has sufficient time to generate the signal.

Qin'19 [82]: It improves on Carlini'18 by using psychoacoustic principles to reduce the perceptibility of adversarial commands. A white-box over-the-air attack is considered. The aim is to generate an adversarial command from an arbitrary input audio example (the two should have similar length) and to reduce the level of distortion noticeable by humans by making use of the masking threshold theory. The generation process is split into two steps. First, similar to Carlini'18, GD is used to find a relatively small perturbation, which causes the perturbed result being transcribed as the target phrase. Then, the second step is applied to make the command imperceptible. The second step also uses GD but with a different function, combining the network loss (cross-entropy loss function) and imperceptibility loss (using the masking threshold knowledge). The work is compared with Carlini'18 and it is shown that the resulting commands are better hidden. The attack is also somewhat more practical as realistic reverberation distortion is simulated. However, the attack is not trialed by playing the adversarial commands through a loudspeaker toward an ASR (an over-the-air attack).

Szurley'19 [83]: It also proposes a method to generate white-box over-the-air targeted adversarial commands based on psychoacoustic properties. To generate commands, the global masking threshold per frame using a psychoacoustic model based on the MPEG-ISO standard is calculated. The distortion level of the perturbation added following the guidance of the psychoacoustic model is measured in the time domain. This results in less computation in each iteration when generating adversarial examples as the calculation can be performed solely in the time domain. Also, this improvement solves the instability issue during backpropagation encountered by Schönherr'19 and Qin'19. The attack signal is also improved for over-the-air attacks by incorporating various room impulse responses (RIRs) generated by a room simulator. The over-the-air evaluation is performed in an anechoic chamber.

Imperio'19 [84]: It describes the first general algorithm to produce robust targeted adversarial commands against the Kaldi ASR assuming a white-box scenario. The adversarial commands generated can be played over the air, and it is reasonably robust even when played in a case where the room characteristics are different than the one used in the signal generation process. This work builds on the implementation of Schönherr'19 by adding another layer simulating the effect of sound being transmitted over the air. This approach maximizes the probability of adversarial examples being transcribed as the target phrase when varying RIR and recording conditions are encountered. The experiments conducted show that more noise has to be added to make the example robust when played over-the-air. In most cases, the attack samples cannot be perfectly transcribed as the target commands. However, it is argued that only one success is required from the attack perspective.

Yang'19 [85]: It proposes a defense method against adversarial commands. This work first evaluates the robustness of input transformation and subsequently temporal dependency-based methods against adversarial commands. Considered state-of-the-art attacks are based on Alzontot'18, CommanderSong'18, and Carlini'18. The experiments show that in general, input transformation methods except autoencoder are effective in defending against Alzontot'18, CommanderSong'18, and Carlini'18 attacks. The work also explores how well a temporal dependency-based defense method works. The principle of such method is: the decoding result when using a portion of an audio signal as input should be similar to the same portion of the decoding result when using the entire audio signal as input. This is due to the temporal dependency (i.e., correlations in consecutive waveform segments). However, perturbation is added to the original audio signal to push the ASR output toward the target content, and thus, the temporal information is lost. The evaluation shows that the temporal dependency method is able to discriminating attacks and benign input generated with methods described in Alzontot'18, CommanderSong'18, and Carlini'18. In an additional step, the work evaluates the two defense methods against adaptive attacks where attackers are aware of the defense mechanisms. The attack by Carlini'18 is used and the results show that input transformation methods fail, while the temporal dependency method represents a successful defense mechanism.

Metamorph'20 [86]: It presents a system called Metamorph that generates robust over-the-air adversarial commands. Like most of the previous over-the-air attack studies such as Yakura'19, Qin'19, Szurley'19, and Imperio'19, this work also incorporates the RIR. In addition, this work uses empirical experiments to analyze how frequency selectivity, which is caused by device distortion, channel effects, and background noise, impacts the attack success rate. The key finding is that the channel effects are the most significant obstacle. When the distance between the speaker and the receiver is long (i.e., >8 m), the channel frequency selectivity effect is dominating and unpredictable. Even though channel and device effects have been considered using RIR, the adversarial examples are not generic enough and cannot adapt to new over-the-air environments. To solve this issue, the generation process is optimized to consider these features. A high attack success rate of over 90% is achieved with a distance of up to 6 m.

AdvPulse'20 [87]: It describes AdvPulse, a practical adversarial audio attack aiming at both speaker recognition and SR systems, altering the recognition results of a streaming audio input in a targeted, synchronization-free and over-the-air manner. In contrast to previous scenarios, this work aims at attacking live-streamed speech. This scenario poses a significant challenge for the attacker as the attack must succeed, while the attacker does not have prior knowledge on the audio input. Almost all existing studies collect an audio clip first and then compute perturbation for the entire clip. However, this work aims at a relatively simple ASR model dedicated to recognizing single-word commands, which makes it easier to overcome the aforementioned obstacle. Targeting this simple model, a 90% success rate in indoor environments and a 70% success rate in an in-vehicle scenario are achieved.

Comparison: A comparison among these studies on adversarial commands (or adversarial examples) is shown in Table 3. The first field is related to the threat and describes the length of the adversarial command being considered in the study. Length describes if the command considered contains only a few words or if it is a long sentence. Practicality captures relevant aspects to be considered when attempting to use the described method in a practical setting. Test method describes how the adversarial command generation method was tested, i.e., was the command submitted over-the-air using a speaker and microphone or was a generated audio file directly fed into an ASR. Room gives a description of the environment in which such over-the-air evaluation was carried out. Distance shows the longest distance between the speaker and the microphone that was considered in case of an over-the-air evaluation scenario. ASR gives details on the used ASR, specifically type, and model. Type shows the key components of the ASR framework used in the study, and model shows the name of the specific ASR used. Context tells if the ASR framework and parameters are known when generating the adversarial commands (referred to as white or black box). Technique shows the key algorithms used in the study to generate the adversarial commands. Perception metrics details the metric used to measure the noise introduced to the original sound sample by the added perturbation. Studies tend to use objective metrics such as SNR as well as launching a user study to test human participants' perception of these adversarial examples. Evaluation metrics name the overall evaluation metrics, describing how well ASRs transcribe the adversarial examples generated with the proposed method. Column perception metrics and evaluation metrics illustrate how an adversarial example is perceived by a human listener

Table 3 Comparison of ASR Adversarial Examples Work (Category 1.2.3)

Papers	Length	Practicality			ASR		Context	Generation Technology	Perception Metrics	Evaluation Metrics
		Test Methods	Room	Distance	Types	Models				
Iter'17	word& three-word phrases & sentences	simulation	N/A	N/A	DNN	WaveNet ASR	white box	FGSM FGM	observation	success rate
Alzantot'18	one word	simulation	N/A	N/A	CNN	Google CNN	grey box	genetic algorithm	user study	success rate
Carlini'18	sentences	simulation	N/A	N/A	RNN -CTC	DeepSpeech	white box	Adam optimisation	decibel& observation	success rate
Commander -Song	sentences	over the air	N/A	1.5 m	DNN -HMM	Kaldi & iFLYTEK	white box	Gradient Descent (GD)	SNR & user study	success rate
Taori'18	two words	simulation	N/A	N/A	RNN -CTC	DeepSpeech	grey box	genetic& gradient	correlation coefficient	CER
Khare'19	sentences	simulation	N/A	N/A	DNN -HMM& RNN -CTC	Kaldi& DeepSpeech	black box	MOGA NSGA-II	Correlation Coefficient & user study	WER
Schönherr'19	sentences	simulation	N/A	N/A	DNN-HMM	Kaldi	white box	GD& psycho-acoustics	user study& MUSHRA	WER
Yakura'19	two or three words	over the air	one room	0.5 m	RNN -CTC	DeepSpeech	white box	Adam optimisation	SNR & user study	success rate & CER
Qin'19	sentences	room simulation	N/A	N/A	RNN & attention	Lingvo	white box	GD& psycho-acoustics	user study	success rate & WER
Szurley'19	one sentence	over the air	one anechoic room	0.16 m no echo	RNN -CTC	DeepSpeech	white box	PGD& psycho-acoustics	SNR PESQ	WER& CER
Imperio'19	sentences	over the air	various rooms	4.3 m	DNN-HMM	Kaldi	white box	GD& psycho-acoustics	SNRseg	success rate & WER
Metamorph'20	sentences	over the air	one room various locations	6 m	RNN -CTC	DeepSpeech	white box	Adam optimisation	MCD user study	character success rate & transcript success rate
AdvPulse'20	one word	over the air	indoor& in-vehicle	2.7 m	CNN	Google CNN	white box	GD	SNR& observation	success rate & confusion matrix

(should be similar to the original benign command) and is interpreted by an ASR (should be very different from the original transcription or should be the same as the target command). Most research on targeted adversarial examples utilizes success rate (introduced in Section II-D.) Specifically, for Metamorph'20, the transcript success rate equals the success rate used in this survey, while character success rate is the concept applied on a character level.

C. Summary

Access control can be achieved by using VA methods and to some degree by checking the semantics of a voice command. Most PVAs only check simple semantics (e.g., does the command contain "Alexa" in case of the Amazon Echo) and some few PVA use VA (e.g., Siri). However, even in the few cases where VA is used, potential attacks, such as replay or spoofing on this mechanism, are usually ignored. It is also assumed that users would realize if someone is interacting unauthorized with their PVA. In normal circumstances, a user can hear voice commands spoken by an adversary or played by a speaker used by the adversary.

There is a large body of existing work looking at VA, how to attack VA, and methods to detect and prevent such attacks. The main threat to VA is spoofing and the results of the ASVspoof challenge summarize the state of the art on spoofing detection in the general SR context. VMask'20 is the first practical black-box attack targeting an ASV system with a spoofing attack and not a general ASR environment (C1.2.1—acoustic characteristics). Spoofing can also be detected using a second data source (C1.2.2—second factor authentication) either by detecting human vocal system features (VoiceLive'16) or traces unique to COTS speakers (Blue'18). While a lot of works exist in the VA space, more studies are required specifically considering the PVAs context.

The area of C1.2—hidden voice commands is a specific domain attracting a lot of works in the research community.

The work described in C1.2.1—hardware nonlinearity requires high-end or highly customized acoustic signal emitters and reported defense work also requires specifically designed speaker arrays. The very recent work EarArray'21 uses a microphone array containing at least three microphones to detect DolphinAttacks relying on the fast attenuation feature of ultrasound [91].

There is not much work on obfuscated commands (C.1.2.2—obfuscated commands) and most of it is early work (before 2017). The reason might be that hiding the malicious voice commands within noise is not covert enough and the research community has shifted to C.1.2.3—adversarial commands.

From Table 3, it can be seen that there is a trend toward generation of adversarial sentences, departing from early work focusing only on words. Also, work is moving on from simple simulation to evaluation of full systems (over-the-air attack on of the shelf PVA). The subject ASRs used are mainly classic DNN-HMM (Kaldi) and end-to-end solutions using RNN-CTC (DeepSpeech). Most of the works are still considering white-box settings where the ASR internals are known to the attacker. Perturbations are generated mainly based on GD optimization. Psychoacoustic masking is increasingly used to optimize addition of perturbations. However, the details on how best to add perturbations are the subject of current work. Most works make use of user studies to evaluate how well adversarial examples are constructed.

More practical and robust adversarial command studies are required, considering evaluation in different reverberation (various rooms or one room but various settings) scenarios. So far, Imperio'19, Metamorph'20, and AdvPulse'20 are three practical works considering this.

Current work lacks practical black-box assumptions; it is always assumed that the internals of an ASR are fully known. However, it might be possible that a variety of systems can be attacked using a common adversarial command. CommanderSong'18 investigated this transferability and a very recent work called Devil's Whisper'21 [92] proposes to use a local white-box model that roughly approximates the target black-box ASR.

There is also a lack of adversarial command research targeting the latest attention/transformer-based end-to-end ASR introduced in Section II-C with Qin'19 [82] being a notable exception. Considering new ASR systems may enable new attack and defense methods.

Currently, there is a lack of metrics for measuring the perceptual distortion of adversarial commands. Research in this area could help building more effective adversarial example studies. We notice that there is a recent work [93] developing this area.

V. C2—ACOUSTIC DENIAL OF SERVICE

The acoustic channel can be subject to a DoS attack. This form of attack on a PVA has attracted little research so far. However, given that we are increasingly depending on PVAs in our daily interaction with (also critical) computer systems, these need to be considered.

Two categories of DoS attacks have been considered: C2.1—skill market and C2.2—jamming. Attacks in C2.1 aim to manipulate the back-end processing (often referred to as the skill market). Jamming attacks (C2.2) target the audio channel directly, interrupting wake word recognition or SR in general.

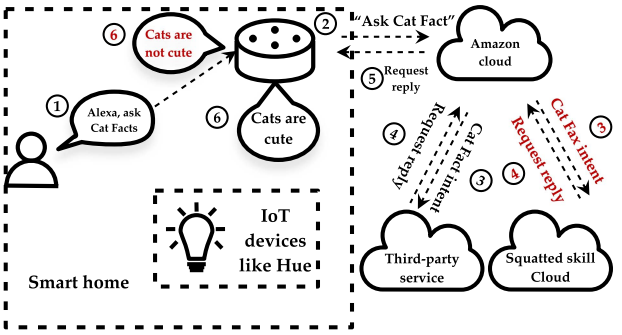


Fig. 6. *Illustration of how a malicious skill created by an adversary launches a skill squatting attack. Steps labeled with black numbers are the normal procedures, while steps labeled with red especially Steps 3, 4, and 6 would replace the original ones when the attack happens.*

A. Skill Market

PVA service and product providers, such as Amazon and Google, provide skills to enrich the capability of PVAs. Users (and service providers) can deploy code to the processing back end, which is activated by a voice command. Skills are a feature that can provide an entry point for an attacker.

Kumar'18 [14]: It presents an empirical study of the misinterpretation error of the Amazon voice recognition service Alexa. Skills are invoked by the back-end infrastructure depending on the transcribed text. Every ASR is subject to interpretation errors and these errors can be exploited to design a skill that is activated by accident when spoken user commands are incorrectly interpreted. Some phrases are likely to be misinterpreted consistently, which can then be exploited to craft a skill that is activated on misinterpretation. This form of attack is called skill squatting (or squatting attack). Fig. 6 shows an example of a malicious skill taking advantage of the misinterpretation to squat attack the legitimate "Cat Fact" skill. The study finds 381 unique instances in which skills invoked by a user might accidentally trigger an already existing different skill. The work also shows the feasibility of squatting attacks toward certain groups of individuals and label this technique spear skill squatting.

Dangerous Skills'19 [94]: It also looks at skill squatting attacks. However, in addition to these attacks based on misinterpretation errors, referred to as voice squatting in this work, additional word squatting and voice masquerading attacks are investigated.

Systems today apply a longest string matching strategy to identify which skills are being called by the users. Thus, word squatting is possible by registering an attack skill with the name of a legitimate skill together with an additional utterance such as "please." For example, it is possible to register an attack skill invoked by the sentence "Cat Fact Service Please" in parallel to a legitimate service invoked by the phrase "Cat Fact Service." Due to the longest string match, the attack skill instead of the legitimate skill may be invoked by simply adding the word "please" to the command, which is quite natural.

Voice masquerading attacks are exemplified in two scenarios. Both Google Home and Amazon Echo allow only one active skill and it needs to be terminated before another one can execute. However, users may naively believe that a PVA supports skill switch. Users would ask to activate another skill while interacting with the current one. This opens a gate for a malicious skill to impersonate the desired one and to obtain sensitive information supposed to only be shared with the target skill. Users also depend on responses from a skill to tell them when it has terminated. A malicious skill could fake the termination by playing the audio response but keeps running. Even if the user uses commands such as "stop" or "cancel" to terminate a skill, a malicious skill can ignore these. If a user does not interact with the PVA for a period of time after one round of inquiry and response, a PVA would reprompt the user with an audio signal before terminating it. However, a malicious skill could create an inaudible audio reprompt and try to stay active with the aim of stealing information from the user.

The evaluation presented in this work shows that the different voice squatting attacks are feasible. As countermeasures, the work proposes a skill scanner to inspect skills before publication and a context-sensitive detector to detect the intent of switching skills and faking skill termination.

B. Jamming

The acoustic channel can be subjected to noise, which prevents ASR from functioning correctly. Jamming signals can either be applied continuously or be targeted more selectively to specific parts of an audio signal. For example, it is possible to target jamming toward wake word recognition to prevent a PVA from processing speech. Jamming is often applied for the purpose of privacy management and the work reviewed here could also be classed in Category C3. However, as these works mainly focus on the jamming component and not on privacy management, we have decided to outline this work here.

Cheng'18 [13]: It proposes a reactive DoS jamming method to prevent people's voice contents being recorded and uploaded to the back-end server by PVAs. This solution gives people the capability to stop nearby PVAs owned by others being activated and recording their conversation. A protection jamming device (PJD) device is proposed,

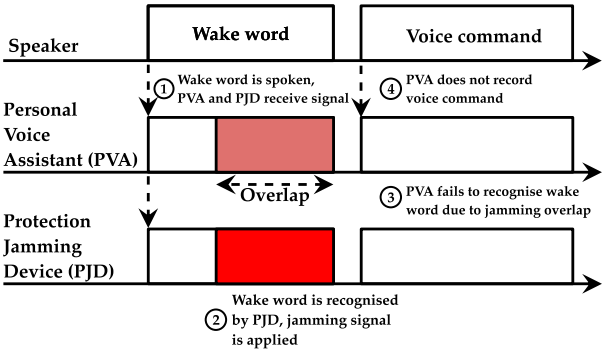


Fig. 7. *PJD recognizes the keyword faster and emits the jamming signal (Step 2) to overlap the rest part of the spoken keyword, and thus, the PVA would not be activated (Step 3) and the following voice commands would not be recorded and processed (Step 4).*

which listens to the same wake word as the PVA (e.g., "Hey Google" for Google Home and "Alexa" for Amazon Echo) recognizes it faster than the PVA and emits a jamming signal to interfere with the wake word acquisition process on the PVA (see Fig. 7). The method is intended to be used as a privacy management, but it is possible to simply use it for DoS purposes.

The work evaluates the impact of key factors of jamming, such as overlap between jamming signal and original audio signal, SNR, and jamming signal types. The experimental evaluation shows that a 100% jamming success and negligible false positive rate (FPR) of wake word recognition can be achieved with an overlap between jamming and audio signal of at least 60%. As it is sufficient to apply a jamming signal only to the latter part of the wake word, it is possible to build a jamming device that emits a signal only after the start of the wake word has been recognized. Such selective jamming devices are hard to detect making the development of countermeasures challenging.

The work Gao'18 [95] differs from Cheng'18 as instead of applying a jamming signal when needed, the channel is continuously jammed, and only when required, the jamming is stopped. The work proposes an external obfuscator, which continuously carries out jamming to prevent conversation recording. Inaudible jamming exploiting the nonlinear property of microphones is used to make jamming user friendly (see also Section IV-B where this technique is used to construct inaudible commands for command injection). An ultrasound microphone installed on the obfuscator is used to listen to wake words. As this microphone type has a linear property within the very high-frequency range, the ultrasound jamming signal sent by the speaker of the obfuscator does not create a shadow in the voice frequency range. In this case, the microphone can capture the wake word and the obfuscator can perform wake word recognition. If the wake word is detected and a challenge–response authentication with the legitimate user's smartphone using a secure out-of-band (OOB) channel such as Bluetooth has been carried out,

jamming stops. Then, the obfuscator communicates the wake word to the PVA via ultrasound injection. Subsequent voice commands will be relayed to the PVA.

MicShield'20 [96]: It is similar to the aforementioned work Gao'18. The main difference is that this work describes a full implementation, while Gao'18 only describes the approach. The OOB channel between the user's smartphone and PVA is omitted; once the wake word is recognized, jamming stops, and the PVA is still able to process the audio signal.

The documented experiments show that the wake word is still recognized if only a few milliseconds at the start of the wake word are subject to jamming. For example, if the first 60 ms of the wake word are jammed, it can still activate an Echo Dot with 95% accuracy, the same as it is the case without jamming.

An implementation of MicShield comprising one ultrasonic transducer, one microphone, and one Raspberry pi [97] is evaluated. The transducer emitting the jamming signal is relatively close to the PVA microphone (14 cm). In a real-world setting, MicShield is able to achieve 90.4% mute rate (duration of the jamming signal is applied to private speech) and 0.02% SR rate for private speech, without affecting the PVA's ability to detect wake words even in noisy environments across different PVA locations.

C. Summary

Kumar'18 and Dangerous Skills'19 focus on security vulnerabilities in the PVA back-end skill working mechanism. Cheng'18 and Gao'18 focus on front-end PVA devices preventing user privacy violation and unauthorized commands via DoS. Kumar'18 studies the interpretation error of the back-end ASR system systematically and discovers the error pattern. Dangerous Skills'19 discovers the same interpretation error, also a skill search issue, and malicious skill impersonation attack. Kumar'18 studies interpretation attacks more deeply, which could be used as an instruction to improve the back-end system, while Dangerous Skills'19 covers wider potential skill-related problems revealing that back-end skill management needs more attention and study. Cheng'18 proposes protecting user privacy using DoS attack targeting wake word of a PVA. It is carried out some early trial to test the feasibility. More comprehensive and systematic implementation of this idea in COTS devices scenario would be of interest. Gao'18 proposes a framework achieving a similar jamming idea as Cheng'18, and they also propose using a user's smartphone together with their obfuscator to achieve legitimate user authentication. This work is only a theoretical framework that is technically complex and it is not sure how practically feasible it is. MicShield'20 is very similar to Gao'18 and provides full implementation and evaluation.

Overall, the potential of applying DoS techniques on the back-end PVA system and the front-end PVA devices has not been fully explored. In particular, DoS attacks on the front end have been used to implement some form of privacy control; a proper analysis of DoS capabilities and potential defense methods is missing. More studies of utilizing DoS for both attack or defense purposes would be valuable.

VI. C3—VOICE PRIVACY

In this category, we summarize work concerned with protecting the privacy of users' voice data. Privacy preservation (Category C3.1—privacy preservation) is a growing research area, in part motivated by the growing popularity of PVAs. Work described here aims to prevent speaker recognition and/or the extraction of paralinguistic information (e.g., features such as emotion, gender, or age) from voice. Some work in Category C3.1 is therefore complementary to the work described in C1.1—VA; here, the focus is on preventing speaker recognition, while in C1.1, speaker recognition is used for authentication. Similar to work presented in C1.1, work on privacy preservation (C3.1) has broad relevance and is only applicable to the PVA context.

We also include consent management (Category C3.2—consent management) as a topic parallel to privacy preservation. This is a new strand of work emerging in the PVA domain. As mentioned before, this work is related to C2.2—jamming as often jamming is employed as a technique to implement consent management; however, additional mechanisms are proposed, which are discussed here.

A. Privacy Preservation

To provide good user experience, PVAs are "always listening," which unfortunately raises privacy concerns. It has been found that voice data also contain rich information about the speaker, including information on gender, age, health, and ethnicity. Voice data can be used for identification (voice biometrics) as discussed in Section IV-A. Voice data are considered sensitive data and legal frameworks, such as the EU GDPR [24], must be considered. For example, GDPR requires that a user gives explicit consent (we discuss this issue in more detail in Section VI-B) and that systems incorporate privacy by design approach. According to GDPR, data controllers should protect users' rights and freedoms in relation to the processing of their personal data; there is an obligation of data protection by design and by default.

Speaker identity (voice biometrics) is usually considered the utmost private information in voice data; however, nonbiometric data also contain a wealth of sensitive and private information that requires protection. A user would like to have assurances (and control) over which elements of biometric and nonbiometric information are processed and shared. In this section, we discuss the work that provides mechanisms designed to limit private information shared via voice data.

If SR would only be performed on the local PVA device, privacy could be preserved to a certain extent. However,

due to resource limitations of these devices, on-device SR is limited. Also, PVA vendors favor a centralized setup where ASR is executed in the back-end infrastructure. Manufacturers are advancing relevant developments [98] and it may be that ASR will be executed completely offline on local devices in the near future. This will give a user potentially more control over private elements of voice data. However, the transcribed text (or other extracted features, depending on the application) will still be passed on to the back-end infrastructure for further processing. A complete local execution of ASR and the following service processing chain is not yet feasible.

Hardware-assisted security for private speech processing has also been explored. An architecture using Intel's SGX architecture called VoiceGuard'18 [99] has been proposed. SGX allows code to define private regions of memory, called enclaves, whose contents are protected and cannot be accessed by any process outside the enclave itself. In VoiceGuard'18, speech characterization is carried out in an SGX enclave and voice data provided by the user are not accessible to the back end, while the back-end models are not accessible to the user. Solutions such as VoiceGuard'18 have limitations; hardware support is required, and in case of SGX, the size of an enclave is limited, which prevents execution of complex tasks (speaker recognition is possible but full ASR is challenging).

The recently initiated Voice Privacy Challenge'20 [100] defines four categories in which voice privacy preservation solutions fall into: deletion, encryption, distributed learning, and anonymization.

Deletion techniques aim at ambient sound analysis. When recording sound in public places, speech is obfuscated such that no information can be recovered [101]. This can be seen as a similar technique to blurring all faces in video surveillance in public places. The work described in C2—acoustic DoS can be seen as another technique to achieve this goal.

Encryption schemes such as homomorphic encryption [102] or secure multiparty computation [99] can be used to process data in encrypted form. Such methods can be transferred into the speaker/speech processing domain to solve some of the outlined privacy challenges.

Cryptographic approaches are surveyed by Nautsch'19 [103]. The work distinguishes between privacy preservation of biometric and nonbiometric data.

Biometric systems use separate enrollment and verification phases. During enrollment, biometric references are collected and features and their representations (templates or models) are extracted and stored. During verification, a probe is captured and compared with the stored representation. It is not desirable to leave enrollment data (speaker models) unprotected as this would allow an attacker to generate speech representative of the speaker. Similarly, probes used during verification should also be protected. Methods exist to provide protection for biometric information, which can be adapted for speaker characterization. Voice representations during enrollment are stored in encrypted form, which still allows comparison with probes collected during verification in the encrypted domain. Possible solutions are based on homomorphic encryption, secure two-party computation, string representation comparisons, and template/model binarization techniques (see [100] and [103]–[105] for examples).

Cryptographic approaches can also reserve privacy in nonbiometric speech characterization applications. For this purpose, similar methods as used for the protection of biometric data can be employed. A user may not want to share an unencrypted speech signal with a back end, while the back end does not want to share a trained model with the user. Methods exist that are based on homomorphic encryption and secure two-party computation [105], [106] or modular hashing [107] to achieve privacy preservation for paralinguistic tasks, such as emotion recognition [107] or detection of voice-affecting diseases [108].

Cryptographic solutions are not yet fully practical for PVA devices. Homomorphic encryption and secure two-party computation schemes are currently limited in the number and size of layers and further improvements are required before these cryptographic schemes can be used to augment existing ASR while achieving the same classification performance. Therefore, cryptographic approaches for privacy preservation in the PVA context are a promising research direction, but they are not yet a practical choice.

Distributed (sometimes referred to as federated) learning methods aim to train models from distributed data without directly accessing it [109]. A decentralized optimization procedure is used to train a central model on local data of many users without the need to upload this data to a central server. As central data collection for training purposes is avoided, some privacy issues can be avoided. However, federated systems still leak information via model updates [110]. Recent work by Granqvist *et al.* [111] therefore combine federated learning with differential privacy. With differential privacy, noise is added to model updates to provide a guaranteed upper bound on the amount of information that can be leaked. Distributed learning technologies improve user privacy, but this approach is focused on the model. It does not change the operation of a system once it is built. In addition, model updating can still leak information of user's private data [110], which indicates that more research in the context of federated learning is necessary.

Anonymization refers to the goal of suppressing (some) personally identifiable attributes of the speech signal while leaving other attributes intact [100]. This approach has attracted the most research effort so far. Existing work includes Hidebehind'18 [112], Gong'18 [113], Abdullah'19-2 [77], Nelus'19 [114], Emotionless'19 [115], and Smart^2 Speaker Blocker'19 [116], which we will discuss in this section.

A common problem for these anonymization solutions is that there is a lack of a formal definition of anonymization and attacks against it. In addition, similar to the case of spoofing detection introduced in Section IV-A1, it is

difficult to compare the proposed solutions due to the lack of common datasets, protocols, and metrics. To address these limitations, the Voice Privacy Challenge'20 [100] was founded in 2020.

The challenge formulates the privacy preservation problem as a game between users and attackers in which users publish data which the attacker can access and attempts to infer personal information about the user from it. For privacy protection, the user publishes as little personal information as possible while allowing one or more desired goals to be fulfilled. The challenge defines the specific task of hiding speaker identities while allowing any other goal (such as SR) to be achieved.

To hide the identity, a user passes utterances through an anonymization system before publication. The anonymized utterances sound as if spoken by another person, which may be an artificial voice not corresponding to any real speaker (pseudo-speaker). The output of the anonymization system is required to be a speech waveform, should hide the speaker identity, should not distort other speech characteristics, and should ensure that all trial utterances from a given speaker appear to be uttered by the same pseudo-speaker.

The attacker has access to anonymized trial utterances and anonymized or original enrollment utterances for each speaker. The anonymization system is a black box for the attacker. The protection performance is assessed via objective speaker verifiability metrics, subjective speaker verifiability metrics, and linkability metrics.

Publicly available datasets, such as VoxCeleb, LibriSpeech, and VCTK, are used for the training, development, and evaluation of the anonymization system. An ASV system and an ASR system are used to assess speaker verifiability and ASR decoding error for objective evaluation. Listening tests with subjective metrics, including speaker verifiability, speaker linkability, speech intelligibility, and speech naturalness, are carried out by the challenge organizers. The organizers also introduce two baseline anonymization systems and their objective evaluation results. Challenge participants can be inspired by these two baselines to improve over them.

While the Voice Privacy Challenge'20 provides a good contribution in terms of evaluation and comparison of anonymization techniques, it has the drawback that it limits the scope of possible solutions. Solutions under this scheme must fit in the outlined framework of the challenge.

In the following paragraphs, we detail existing contributions to the domain of anonymization. Some of these solutions could be evaluated under the scheme set out by the Voice Privacy Challenge'20, while others do not fit in this framework.

Hidebehind'18 [112]: It introduces VoiceMask (the work is titled Hidebehind, while the solution is termed VoiceMask) as an intermediary between the PVAs and the cloud used for SR to anonymize the voice before it reaches the untrusted system, as shown in Fig. 8. In this work, two

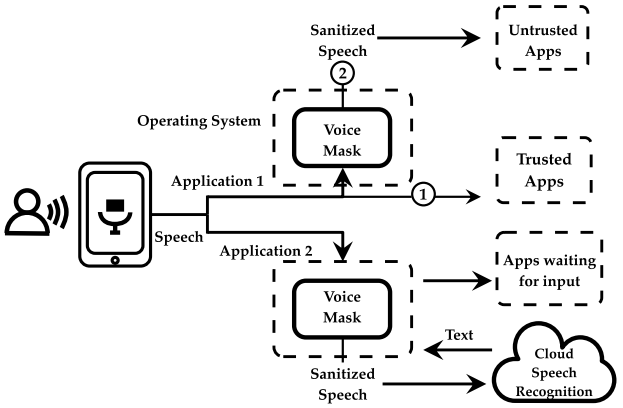


Fig. 8. *Illustration of the two application scenarios of VoiceMask (Hidebehind'18). The first scenario is shown in the upper part. VoiceMask can be embedded in the OS and only sanitizes the speech sent to untrusted Apps that do not need to access the original voice as shown in Route 2. Trusted Apps that must obtain the original voice are granted Route 1. The second scenario is shown in the lower part. VoiceMask is used as gateway between user speech and Apps and cloud. VoiceMask masks the original speech and sends it to an online ASR interface. After obtaining the transcript, they are relayed to the Apps.*

warping functions together with a differential privacy algorithm are used to construct a robust conversion algorithm, which reduces the success rate of speaker recognition. The method provides resilience against deanonymization attacks while ensuring that the resulting signal is still recognizable.

Gong'18 [113]: It aims to distort a speech signal such that paralinguistic feature detection fails, while the signal distortion is difficult to recognize by a human. The work proposes an end-to-end scheme to craft adversarial audio signals starting with the original audio signal rather than with already extracted acoustic features. These adversarial signals result in a performance drop of state-of-the-art NNs (specifically, Recurrent Neural Network (RNN) and convolutional neural network (CNN)) used for speech paralinguistic analysis. By adding perturbations directly to the original audio signal instead of adding these to acoustic features such as MFCC parameters, human noticeable signal distortion is reduced. Although the authors did not describe the method as a privacy-preserving mechanism, we decided to place it in this category within this survey. Similar to the previously described work Hidebehind'18, this method could be used to hide voice features from a PVA.

Abdullah'19-2 [77]: It uses obfuscation techniques as also used in works described in Section IV-B2. However, instead of modifying an audio signal such that a machine is able to understand the command but a human is not, the signal is modified such that an ASR is unable to understand the command but a human is. Thus, the method is an anonymization solution and can be used to improve a user's privacy. The audio signal is decomposed and components for which the signal strength is below the human perception thresholds are discarded. The reason

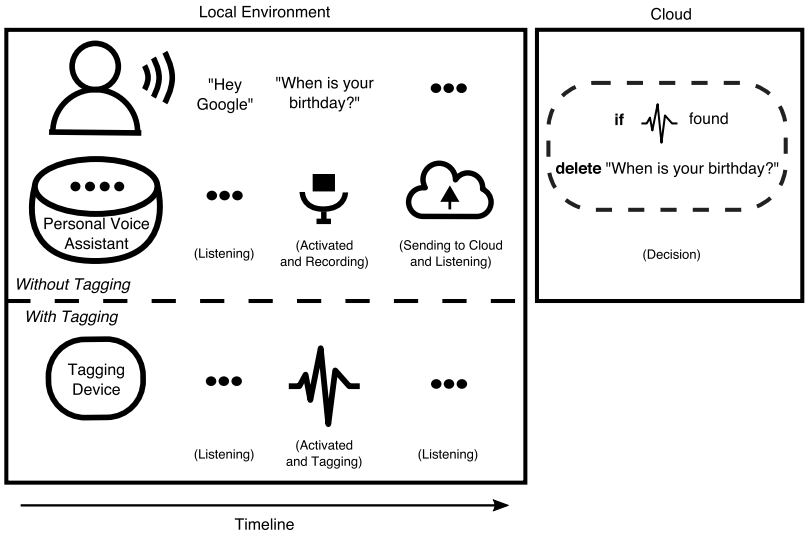


Fig. 9. *Illustration of acoustic tagging as described in Cheng'19. When a tag is found to be embedded, the PVA service provider can delete a recording as consent was not given.*

behind this approach is that low-intensity components should not affect the audio quality perception of humans. However, the performance of the ASR and automatic voice identification (AVI) is affected given the assumption that ASR and AVI depend on components of speech, which are noncritical for human perception.

Nelus'19 [114]: It proposes a feature extraction scheme, which selectively suppresses the biometric information contained within the speech signal but tries to maintain a good gender discriminating performance at the same time. It is assumed that an attacker compromises this gender discriminating system by intercepting the feature set and then aims to perform speaker recognition. The feature extraction process is modified such that a good gender discrimination accuracy is achieved by minimizing the cross entropy between the gender labels' true probability and the estimated probability distributions. In addition, it ensures a bad classification accuracy for tasks such as speaker recognition by minimizing the information in the high-level feature set. The results show that biometric privacy risks can be significantly reduced with only a slight decrease in gender discrimination performance (utility).

Emotionless'19 [115]: It proposes a privacy-preserving intermediate layer between users and cloud services to sanitize the voice input. This work uses emotion states as the example of the sensitive part of a speech signal and aims to normalize it while preserving the signal utility (e.g., speech content and identification information in the voice input) before sending it to the cloud. cycle generative adversarial network (CycleGAN) is used to transfer the original voice input to an "emotionless" speech signal. First, unsupervised learning is used to extract the representation from the speech. Then, a feature extraction model is trained to identify the sensitive features from the speech and convert them to nonsensitive features. A computed feature is used to train a feature extraction model to extract these specific features. Feature conversion is finally applied to hide the sensitive part in the data. The resulting nonsensitive features are synthesized to generate speech.

Smart^2 Speaker Blocker'19 [116]: It proposes a physical infrastructure to improve control over the signals reaching the PVA. The PVA is placed in a soundproof enclosure together with a speaker. The speaker is connected to a protection device called the smart speaker blocker. The smart speaker blocker is essentially a second PVA, which is controlled by the user and is used to filter voice commands before they reach the main PVA.

The smart speaker blocker uses its own voice recognition system based on Sphinx4. Depending on filter rules set on the smart speaker blocker, actions are taken. If a voice command is accepted to be passed to the main PVA, a TTS module is used to transfer the command to the PVA in the enclosure.

As the original voice is not passed to the PVA in the enclosure, it is ensured that speaker recognition cannot be carried out. Also, other cues, such as emotions or health information, cannot be extracted from the speech signal. Furthermore, it can be tightly controlled in which speech signals reach the PVA; accidental overhearing of conversations can be prevented.

B. Consent Management

Users usually have little or no control over PVAs in their vicinity. A user can prevent his own device from recording but would not be able to stop other devices from recording conversations.

A user can prevent a PVA from recording voice completely by simply disabling the device. Such methods have been proposed and we discuss these in Section V-B as they are effectively DoS attacks; DoS is employed as a method of recording control.

For more fine-grained control other methods than DoS are required. Generally, methods to signal recording consent of individuals require the cooperation of the back-end infrastructure. Speaker recognition for access control purposes might be usable for implementation of a recording consent management system, but such options have not yet been explored in the literature.

Legal frameworks, such as the EU GDPR [24], also require that a legal basis exists for processing of personal data and this usually requires that users have given explicit consent. It is also debated if consent management is something that the device manufacturer has to solve as the manufacturer is usually the data controller and processor or if even the user operating a PVA at home is considered a controller with respect to visitors to his home [117]. The European Data Protection Supervisor (EDPS) published a recent note (TechDispatch) [118] pointing out that "speakers can mistakenly detect a spoken expression as their wake-up expression and therefore process personal data without user consent" and that there is "a lack of an appropriate consent management mechanism."

In the context of PVAs, compliance with legal requirements is currently difficult to implement as anyone in the vicinity of a device will be recorded and the technical means to request user consent are nonexistent.

Cheng'19 [46]: It explores acoustic tags as means of recording consent management. Acoustic tags can be embedded with audio signals to indicate to a cooperating back-end infrastructure that a user does not give the consent of recording their voice. In addition, the embedded tag signal can carry information such as when and where the conversation was recorded. Fig. 9 shows how the tagging system works in a PVA usage scenario. It is shown that the tag can also be placed in a frequency band outside of the audible frequency range. The researchers experiment with an off-the-shelf Google Home Mini to explore acoustic recording characteristics (e.g., sampling frequency) of the PVA and the back-end system. Based on the knowledge of these characteristics, they customized a Raspberry pi [97] to be a reactive tagging device, which listens to the wake word and emits a tag signal once activated.

Sigg'20 [119]: It proposes to establish a trust zone through audio fingerprinting, which is then used for consent management. It is assumed that a user has an audio-capable device such as a phone. An audio signal, such as a voice command, is recorded by the PVA and by the user's personal device (PD). A shared master secret K_{MS} is derived from the recorded audio signal and a trust zone is established. Only PVAs within "hearing distance" can obtain the shared master secret K_{MS}. Following this, Diffie–Hellman authenticated key exchange is executed between the PD and any PVA within the trust zone to derive a public/private key pair (using other means than the acoustic channel). K_{MS} is used in this exchange to validate the presence in the trust zone. The PVA now uses the private key to sign the audio signal when it is shared with other devices or the back-end infrastructure. K_{MS} and the public key are kept so that it is possible to later verify that an audio recording was obtained from a trust zone and was provided with consent.

The work requires collaboration of the PVA infrastructure. The work also does not elaborate in detail how key material should be used in a large-scale infrastructure to manage and verify consent. The PD requires interaction with the acoustic channel and exchange of additional information via the network, making this approach more complex than the approach described in Cheng'19. This work marks audio recordings as having explicit consent, while Cheng'19 proposes to tag recordings for which no consent was given.

C. Summary

Work aiming at privacy preservation (Category C3.1) has considered hardware-assisted solutions, deletion techniques, federated learning, cryptographic approaches, and anonymization techniques. Hardware-assisted solutions are possible, but this option has not been explored much. Deletion techniques are not sufficiently selective as all acoustic processing is blocked. Federated learning is an option to prevent centralized data collection, but this technique only addresses privacy aspects of models. Cryptographic techniques are a promising avenue to address many privacy issues in a PVA context. However, cryptographic approaches for privacy preservation in the PVA context are not yet a practical choice; performance and usability of algorithms must be improved.

The vast majority of current solutions for privacy preservation employ anonymization techniques. The goal is to suppress some personally identifiable attributes of the speech signal while leaving other attributes intact. The recent creation of the Voice Privacy Challenge has defined a standardized framework to evaluate works in this space. While the framework makes solutions comparable, it defines a narrow setting in which not all existing solutions will fit.

Table 4 provides a comparison of anonymization solutions detailed in this work. Aspects considered are: the target point in the PVA ecosystem (target), the purpose of the protection method (purposes), and the main techniques used for the approach (technology).

Hidebehind'18 uses VC techniques combining extra safety measures to hide the voiceprint of users. It is a promising research direction and there is room for efficiency improvements as outlined in the work.

Gong'18 aims to protect paralinguistic information by adding unnoticeable adversarial examples to the speech signal. Utilizing adversarial examples for protecting privacy of both paralinguistic and identity (biometrics) information in speech is an emerging study area. A common weakness of this technology is the practicality, and this work is no exception.

Abdullah'19-2 shows how to modify a speech signal such that an ASR is unable to understand the command but a human is. Although the application scenario of this article

Table 4 Comparison of Privacy Preservation Work Using Anonymization. In the Target Column, We Use the Term "Presumably" If This Is Not Explicitly Mentioned in the Work but It Can Be Inferred

Work	Target	Purposes	Technology
Hidebehind'18	smart devices	concealing speaker voiceprint	voice conversion & differential privacy
Gong'18	smart devices & cloud	downgrading paralinguistic analysis performance	generating adversarial examples with FGSM
Abdullah'19-2	smart devices	downgrading ASR & AVI performance	DFT & SSA
Nelus'19	N/A	downgrading speaker recognition & maintaining gender discrimination	variational information feature extraction
Emotionless'19	presumably on smart devices	downgrading emotion identification & maintaining other unitily	classic speech processing & CycleGAN & speech synthesis
Smart^2 Speaker Blocker'19	smart speakers	filtering speech content & paralinguistic information	speech recognition & TTS

is antisurveillance in telephony networks, the work should not only be limited to this case.

Nelus'19 designs a feature extraction scheme to suppress the speech identity information but maintain a good gender discriminating performance. The idea of designing a specific feature extractor, which satisfies certain purpose, suppressing unnecessary but sensitive elements in speech signal, is interesting. This idea could really shine if the proposed system can be deployed on users' devices, giving users the capability of controlling their own privacy. This approach is potentially more lightweight compared to cryptography-based solutions. Unfortunately, the practicality of the proposed framework is not discussed in this work.

Emotionless'19 captures the key features representing emotion states and then converts the signal to be emotionless using CycleGAN while aiming to maintain speaker and SR. However, a 35% accuracy drop is high and improvements are necessary. The work considers deployment on user end devices, and however, there is no evaluation on the practicality of this approach.

Smart^2 Speaker Blocker'19 is effective in protecting biometric and nonbiometric information but is sacrificing usability due to the inclusion of an additional PVA, TTS module, and customized hardware.

There is little work in the consent management Category C3.2. As discussed, DoS methods can be employed to implement a simple form of consent management by blocking all PVA activity (see also Category C2). We only found two works (Cheng'19 and Sigg'20) that specifically address this privacy aspect. Given that many legal frameworks such as the EU GDPR require explicit consent of users, it is surprising to find how little research effort is directed toward technical solutions for consent management.

VII. C4—ACOUSTIC SENSING

In this category, we describe work that uses an acoustic channel to perform sensing. We distinguish C4.1—passive sensing and C4.2—active sensing. For active sensing, the PVA can be used to emit a sound signal and reflections are analyzed for sensing purposes. Passive sensing relies on analysis of sound not specifically emitted for the sensing task itself.

Work in this category uses sound for sensing but in most cases not in the context of a PVA. In many cases, a mobile phone, which may include PVA components, is considered. However, dedicated smart speaker devices have not yet been investigated. A number of works here aim to infer user interactions with a phone/tablet using sound (e.g., revealing user input). Nevertheless, the approaches discussed here can be applied to the PVA domain and they provide valuable insight into security and privacy of PVAs.

A. Passive Sensing

Sound signals can be analyzed to obtain information about the environment. Any object, machine, or person in the vicinity of a microphone that emits sound can be used to reveal information about this entity. In the context of this work, we are interested in the extraction of security and privacy-relevant information using this approach.

Existing work in this category has a strong focus on analyzing sound cues to infer user movement and actions. A large body of work aims at inferring user interaction with (virtual) keyboards using sound to reveal personal identification number (PIN) numbers. The works Soundcomber'11 [120], Narain'14 [121], Liu'15 [122], and Hearing Your Touch'19 [123] are examples here. Other works

aim at revealing user movement more generally, such as location of people or the presence of a specific person (Shen'20 [124] and Diapoulis'18 [125]).

A smaller body of work aims at using sound to obtain information about the state of machines. For example, sound emitted from computers has been used by Genkin'14 [126] to reveal cryptographic keys. Synesthesia'19 [127] uses emitted sound to reveal display content.

Speech analysis could also be classified as a form of acoustic sensing. However, here, we include work that looks at information included in a sound signal other than voice.

Soundcomber'11 [120]: It proposes a covert smartphone attack to steal information such as credit card numbers or PIN numbers. The work assumes that malware components are installed on the phone. Soundcomber has access to the microphone during calls and has tone and SR components. When keywords are recognized at the beginning of the call, keyboard tone inputs and call segments are analyzed to identify high-value elements such as credit card numbers. Narain'14 [121] studies the feasibility of inferring keystrokes on virtual keyboards of an Android smartphone. Information extracted from microphones and gyroscope of the phone is analyzed; a model is built using training data. Attack performance is best when combining data from both sensor sources. Liu'15 [122] introduces a keyboard keystroke snooping attack using the stereo recording from two microphones of a single smartphone. This attack does not require training as in Narain'14 to label keystrokes and exploits TDoA and unique acoustic MFCC features of keystroke sound to achieve millimeter accuracy. The monitored keyboard is a real keyboard (not a virtual one displayed on a screen) external to the phone. Hearing Your Touch'19 [123] proposes the first acoustic side-channel attack revealing the input typed on a virtual keyboard of a touch screen on a smartphone. The sound waves from the finger touching the virtual keyboard travel through the screen surface and the air. This acoustic signal can be captured by built-in microphones on devices (see Fig. 10). The distortions of the sound wave are related to the tap location on the screen and analyzing the recorded acoustic signal can reveal the typed input. The work uses preprocessed TDoA and the cepstrum of the first 128 samples of the audio data acquired by microphones after a tap on the virtual keyboard as main features. These features are input to an linear discriminant analysis (LDA) classifier for keystroke prediction.

Diapoulis'18 [125]: It describes how individuals can be identified using sound recordings of people walking on a wooden floor. The acoustic event (walking sound) is detected by estimating the beginning of transient sound. For each event (onset), features are extracted. Two synthetic feature subspaces are created by applying principal component analysis (PCA). Finally, LDA is performed to classify which individual the event belongs to. The work shows that a PVA can use sound cues in general to infer user behavior in the vicinity of a device.

Shen'20 [124]: It shows how a speaker can be localized using a voice signal and a standard PVA. Based on the fact that a PVA is usually placed near a wall with power outlet, this work develops a PVA-tailored angle of arrival (AoA) algorithm. Both the AoAs and the parameters of the wall are used to calculate the sound source location. The algorithms are developed to achieve localization of arbitrary sound signals and also to deal with the limited acoustic capabilities of the PVA; the PVA used in this work does not provide a sophisticated microphone array normally necessary for accurate acoustic localization. The system achieves 0.44-m accuracy in different settings. This work demonstrates that a PVA is generally able to locate sound sources within a room.

Genkin'14 [126]: It introduces a novel cryptanalysis side-channel attack via acoustic emanations. In this work, a full 4096-bit RSA decryption key is extracted from laptops, using the noise generated by the electronic components of the computer executing decryption of chosen ciphertexts. The noise emitted from the voltage regulation circuits is related to computing activities as the power draw of the CPU varies dramatically depending on the execution patterns of the running algorithms. Such acoustic sensing task can be carried out by a PVA. This highlights the sensing opportunities a PVA listening continuously with high-quality microphones to the acoustic channel has.

Synesthesia'19 [127]: It introduces a novel side-channel attack method to reveal display contents based on the acoustic emanation from the electronic components of the screen. It is found that the spectrogram of the acoustic emanation is related to the pixel period, and the brightness of pixel lines is inversely related to the amplitude of the filtered acoustic signal. Signal processing algorithms are designed based on these observations. The work showcases different attack scenarios with the help of applying machine learning classifiers: detecting user input on the virtual keyboard on the screen, detecting on-screen texts, website fingerprinting attacks (detecting which website is being shown), and voice over Internet protocol (VoIP) attacks inferring if the user is watching the video call window or browsing the web. This work shows that a PVA can potentially use sound cues to sense how users interact with electronic devices.

B. Active Sensing

Active sensing is generally used to obtain information about the position of objects or people. The PVA is used to emit a sound signal and reflections are analyzed for sensing purposes. Active sensing enables to obtain information with much greater detail than is possible with a passive approach. For example, work has shown that it is not only possible to locate people and objects, and it is also possible to obtained details such as gestures and even breathing patterns of a person.

PatternListener'18 [128]: It focuses on observing unlock patterns used on Android phones. Imperceptible sound is sent and recorded with a microphone. The work uses

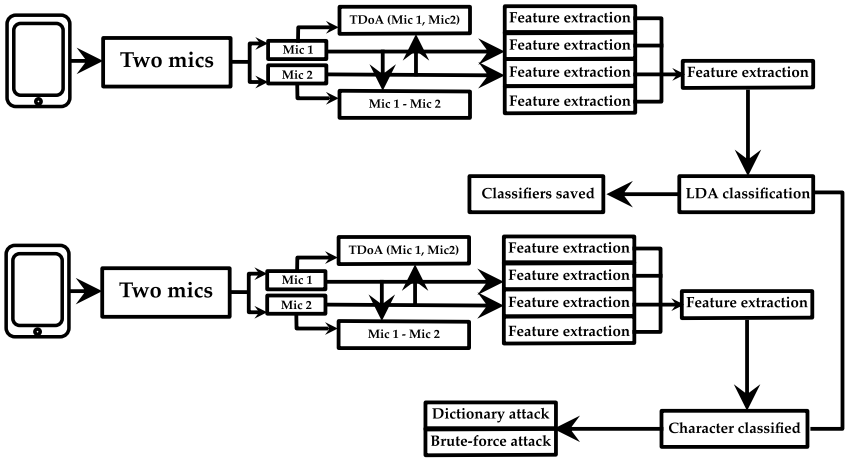


Fig. 10. *Framework of Hearing Your Touch'19: stereo recording using two mics on the phone to record sound when tapping on the virtual keyboard happens. TDoAs for each tapping are calculated. Also, the cepstrum of the signal recorded by either mic is calculated. These two features are used to train an LDA classifier. This classifier is later used for prediction when features of the test data are fed into it.*

a finger movement detection technique introduced by Wang *et al.* [129], which detects phase changes in the emitted sound signal. In addition, motion sensors on the smartphone are used to detect click actions on the screen. The work is similar to Narain'14 as sensing information from the acoustic channel that is combined with additional sensor data. However, active sensing is used here in the acoustic channel. SonarSnoop'18 [6] is similar to PatternListener'18; however, only active acoustic sensing is used. In addition, the used sensing signal is analyzed differently, and instead of phase changes, several features of the received echo profile are used. An inaudible acoustic signal is transmitted through the built-in speakers once a victim draws the unlock pattern. The recorded echoes are used to profile the finger movement composing the unlock pattern with the help of signal processing techniques, such as correlation and Gabor filter and machine learning algorithms. The workflow and additional details are shown in Fig. 11. These works show that a PVA device can function as a sonar system.

CovertBand'17 [130]: It examines the use of a covert acoustic sensing method using music to track human presence and movements. Also, their method can differentiate different types of motion such as pumping arms, jumping, and supine pelvic tilts based on the various spectrograms shown in the echoes. Correlation is used to construct the echo profiles and differentiate moving objects from static objects, and by doing so, it is possible to locate moving individuals. Based on this and further analysis of location, movement tracking and motion classification from multiple targets are feasible. This work shows that normal PVA operations, for example, playing music, can be used to disguise an active sensing signal that collects fine-grained environmental information.

BreathListener'19 [131]: It applies active acoustic sensing to detect driver breath pattern in a noisy car environment. Energy spectrum density (ESD) is utilized to capture all movements in the environment. Further

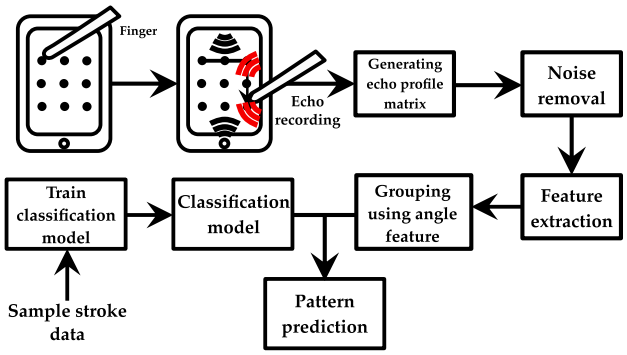


Fig. 11. *Workflow in SonarSnoop'18: inaudible signals are emitted and echoes are recorded, while the unlock pattern is drawn. Echo profile matrix is calculated from the recording using digital signal processing (DSP) techniques. Features including angles and range are extracted from the signal after noise removal. Then, grouping strokes using the angle features and strokes are further predicted using both angle and range features, and thus, patterns are identified.*

interference (movements not related to breath) cancellation is performed by background subtraction and ensemble empirical mode decomposition (EEMD), resulting in ESD signal mainly containing breathing information. It is then transferred to the Hilbert spectrum format. A generative adversarial network (GAN)-based deep learning architecture is used to generate a breathing waveform. This active sensing work showcases the richness of information extraction active acoustic sensing can achieve.

C. Summary

Acoustic sensing is an area of active research. However, few works are directly dedicated to security and privacy in the PVA context. Also, there is less work on active sensing than passive sensing.

Inferring user interaction with a keyboard is one main line of study in the security context as it relates to PIN or password entry. For passive sensing, we include PIN

Skimmer'13, Narain'14, Liu'15, and Hearing Your Touch'19 as representatives as they infer keystrokes on a real/virtual keyboard utilizing microphone recordings on a smartphone, which fits our PVA definition. For active sensing, PatternListener'18 and SonarSnoop'18 are included. They both implement virtual keyboard active sensing using smartphones and showcase stealing unlock patterns.

Apart from keystroke prediction, acoustic sensing using PVA can reveal other information too as the acoustic signal can not only carry verbal messages but also other environmental information. In passive sensing, Soundcomber'11 attacks credit card secrets using call recordings, and Genkin'14 and Synesthesia'19 use sound recordings on smartphones to reveal cryptographic keys and even the monitor's display content. Diapoulis'18 identifies individuals using walking sounds, and Shen'20 locates sound sources using a PVA. In active sensing, CovertBand'17 shows that physical activities can be differentiated using acoustic signals, and health data such as the breathing pattern is extracted in BreathListener'19.

Work in this section showcases what information can be extracted from the acoustic channel and to what extent it can be used. As not many works are directly related to the PVA context, we would like to see more studies making use of PVA acoustic sensing. Acoustic sensing using a PVA is of interest as modern PVA devices provide sophisticated microphone and speaker arrays and additional hardware does not have to be deployed for an attack. Furthermore, PVAs provide sophisticated processing capabilities (on the device and back end) an attacker could harness. Also, devices are already deployed at a large scale and do not raise suspicion. Finally, users expect sound to be emitted from these devices enabling an attacker to conceal active sensing more easily. Existing work on acoustic sensing and security has not exploited these unique conditions to the full. A wide range of interesting security and privacy issues may not have been explored yet.

VIII. CONCLUSION AND DISCUSSION

PVAs are now commonplace and are significantly changing the way users interact with computer systems. Users increasingly depend on PVAs as main or even single interface to computer systems and smart environments. Consequently, security of these devices has become the focus of public attention and research efforts. Likewise, privacy is a concern for most users as PVAs record and observe speech, the most fundamental form of human interaction.

A. Research Challenges

The survey points to a number of open (research) questions in the PVA domain. In the following, we outline what we believe these open questions are and we order them according to our perceived importance.

Access control has received considerable media attention. Users are aware that potentially any user may interact with their PVA and that even commands embedded in songs or adverts played over the radio may be used to interact with their system. However, it seems that pressure from the public is not yet forceful enough to encourage PVA manufacturers to adopt elements of the large body of research work. Therefore, we believe that it is just a matter of time before existing VA techniques will become commonplace. In this scenario, spoofing detection will become important. While there is a long history of work, which started well before the rise of PVAs, it is still essential to adapt these techniques to a large-scale and distributed PVA infrastructure. Thus, the following question should be answered by the research community:

1) How Can Spoofing Detection Be Best Integrated in a Practical PVA Deployment Context?: While command injection is well understood, appropriate defense mechanisms against it require more research work. Current PVAs do not provide any mechanism protecting against this powerful form of attack. We believe therefore that the following question deserves attention.

2) What Are Suitable Protection Methods Against Hidden Commands?: Users also have privacy concerns and this issue has attracted media attention. There is a large body of work outlining methods for privacy preservation, and however, these have not yet found their way into existing PVA deployments. Therefore, we believe that the following question should be answered:

3) How Can Privacy Preservation Methods Be Embedded in a Practical PVA Deployment Context?: Another aspect of voice privacy is consent management. Legal frameworks, such as the EU GDPR and the United States CCPA, require, among other elements, consent when voice is recorded. However, very few works have provided (practical) tools directly intended to implement such consent management.

4) What Are Appropriate Technical Means for the Implementation of Consent Management in the PVA Domain?: PVAs are increasingly used as interfaces for critical systems. For example, PVAs are considered to control equipment in an operating theater or to be used to control features in a car while driving. If used in critical settings, availability must be ensured and it is vital to consider the threat of DoS attacks. However, so far, this area of research is underexplored. The scope of DoS attacks has not been fully explored and appropriate countermeasures do not exist.

5) What Is the Scope of Acoustic DoS Attacks and What Are Suitable Protection Methods Against These?: Users are currently very conscious about cameras being used in private spaces. It is common for users to employ a mechanical cover on a laptop camera when not in use in order to prevent a hacker taking control of the camera. However, this public perception seems not, in general, to apply to sound systems. Most PVAs do not provide a mechanical switch to disconnect speaker or microphone and, where present (as in the Google Home Mini), users rarely make use of it. Users are aware that the device may listen into a conversation, but they are unaware of the highly sophis-

ticated acoustic sensing that is possible. There is a lack of understanding of acoustic sensing in the PVA context and possible defense methods are lacking. Therefore, we believe that the following question should be answered.

6) What Are Security and Privacy Implications of Acoustic Sensing and How Can Users Detect or Defend Against It?:

B. Specific Research Directions

In this section, we outline specific research questions that we believe should be addressed in each of the categories we presented in this survey.

C1.1.1—Acoustic Characteristics: VA is a well-researched area and it is possible to authenticate based on a voice profile. However, as this survey has shown, it is necessary to consider VA more specifically in the context of PVA instead of a general speaker recognition system. Research work needs to consider how to apply VA in current PVA and how to protect specifically against spoofing attacks in this context.

C1.1.2—Second Factor Authentication: We believe that work using a second factor for authentication is promising, in particular when this information can also be derived from the acoustic channel. This approach makes such solutions very practical. Although some works exist describing second factor authentication, research has not looked into bypassing such methods. This is a crucial step to ensure that such methods are robust against attacks.

C1.2.1—Hardware Nonlinearity: Research has shown that such attacks are very feasible. However, there has not been much work on defense mechanisms against this type of attack. Furthermore, attacks (and the few reported defense mechanisms) rely on sophisticated hardware. We would like to see whether the requirement for additional equipment can be overcome.

C.1.2.2—Obfuscated Commands: Feasible attacks are described. However, this form of attack is not very convincing as noise will still be noted. For this line of work, more detailed studies on perception of audio samples are required to fully determine the feasibility of these attack types.

C.1.2.3—Adversarial Commands: This line of work has produced very sophisticated attacks. Complex sentences can be embedded in audio samples, hidden from users. Psychoacoustic masking is increasingly used and attacks over the air considering room characteristics are feasible. However, most attacks still consider white-box scenarios where the internal structure of the ASR is known. Work should consider black-box scenarios and investigate how to craft hidden commands effective on different ASRs. This field of work would benefit from a standardized evaluation environment to make attacks comparable. Research here should target the latest attention/transformer-based end-to-end ASR. Some attacks require significant computation and time to produce an attack signal; more efficient methods are necessary to create attack signals on the fly. Finally, it is necessary for further research on defenses against such powerful attacks.

C.2.1—Skill Market: Multiple works identified in this survey show that the operation of the skill market represents an attack surface. The mapping between voice commands and actions can be exploited by an attacker. Transcriptions of speech are subject to errors, which can be exploited. However, a full-scale systematic misinterpretation analysis is yet to be completed followed by work proposing suitable defense mechanisms.

C.2.2—Jamming: Jamming of PVAs via the acoustic channel is feasible. Noise can be added to prevent a PVA from functioning. Existing work does not use sophisticated jamming methods (i.e., inaudible jamming, jamming preventing detection, and localization). Also, jamming so far had the aim to block a signal entirely; however, it might also be possible to add very targeted interference to introduce more subtle ASR transcription errors. Defense methods to detect jamming or to design PVA resilient to jamming are missing.

C.3.1—Privacy Preservation: There is a body of work considering anonymization techniques. Recently, the Voice Privacy Challenge has been set up to standardize evaluation of this work. Although such methods are effective, it is not clear how they can be integrated with existing systems and how a user would exercise control. Cryptographic methods are promising to preserve privacy; here, these techniques need to advance to make them a feasible option for a practical PVA context.

C.3.2—Consent Management: Only a few works have so far investigated how users can provide consent. Some work on DoS has been carried out as mechanism of revoking consent. We believe that this would be an important area for users and that more research in this domain is required.

C.4.1—Passive Sensing: Works identified in this survey show that the acoustic channel can provide a rich set of information in addition to speech. The acoustic channel has been extensively used to infer user interaction patterns with devices (mainly interaction with phones). It has also been shown that a wide variety of other user behavior, such as laughter, crying, or eating, can be inferred [5]. However, a detailed analysis of what information can be extracted via a PVA is missing. Also, no defense mechanisms against the use of a PVA as an acoustic sensor has been reported.

C.4.2—Active Sensing: The work in this category is similar to the line of work on passive sensing. However, as active signal generation is used, more detailed information can be obtained. It has not yet been investigated in detail how active sensing can be carried out on smart speaker type PVA, work so far has focused on phone-based PVAs. Specifically, how an active sensing signal can be hidden or embedded in expected audio signals (hidden sensing) has not attracted work. For example, sound (voice, music, and so on) emitted from a smart speaker could be designed such that it functions well as an active sensing signal too. Work on how to detect or defend against an active acoustic sensing signal has not yet been explored. ■

REFERENCES

[1] (2019). *Nest Secure's Control Hub has a Microphone—Users Only Found Out When it Became Google Assistant Enabled This Week*. Accessed: Sep. 16, 2019. [Online]. Available: https://voicebot.ai/2019/02/07/nest-secures-control-hub-has-a-microphone-users-only-found-out-when-it-became-google-assistant-enabled-this-week/

[2] NPR. (2018). *The Smart Audio Report*. Accessed: Aug. 3, 2019. [Online]. Available: https://www.nationalpublicmedia.com

[3] Pew Research Center. (2021). *Mobile Fact Sheet*. [Online]. Available: https://www.pewresearch.org/internet/fact-sheet/mobile/

[4] J. Lau, B. Zimmerman, and F. Schaub, "Alexa, are you listening? Privacy perceptions, concerns and privacy-seeking behaviors with smart speakers," *Proc. ACM Hum.-Comput. Interact.*, vol. 2, pp. 102:1–102:31, Nov. 2018, doi: 10.1145/3274371.

[5] T. Rahman *et al.*, "BodyBeat: A mobile system for sensing non-speech body sounds," in *Proc. 12th Annu. Int. Conf. Mobile Syst., Appl., Services (MobiSys)*, New York, NY, USA, Jun. 2014, pp. 2–13, doi: 10.1145/2594368.2594386.

[6] P. Cheng, I. E. Bagci, U. Roedig, and J. Yan, "SonarSnoop: Active acoustic side-channel attacks," *CoRR*, vol. abs/1808.10250, pp. 1–13, Aug. 2018.

[7] W. Diao, X. Liu, Z. Zhou, and K. Zhang, "Your voice assistant is mine: How to abuse speakers to steal information and control your phone," in *Proc. 4th ACM Workshop Secur. Privacy Smartphones Mobile Devices (SPSM)*, New York, NY, USA, Nov. 2014, pp. 63–74.

[8] G. Zhang, C. Yan, X. Ji, T. Zhang, T. Zhang, and W. Xu, "DolphinAttack: Inaudible voice commands," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, New York, NY, USA, Oct. 2017, pp. 103–117, doi: 10.1145/3133956.3134052.

[9] L. Schönherr, K. Kohls, S. Zeiler, T. Holz, and D. Kolossa, "Adversarial attacks against automatic speech recognition systems via psychoacoustic hiding," in *Proc. Netw. Distrib. Syst. Secur. Symp. (NDSS)*, 2019, pp. 1–15.

[10] Siri Team, "Personalized hey siri—Apple," *Apple Mach. Learn. J.*, vol. 1, no. 9, Apr. 2018. [Online]. Available: https://machinelearning.apple.com/research/personalized-hey-siri

[11] R. Zhang, X. Chen, J. Lu, S. Wen, S. Nepal, and Y. Xiang, "Using AI to hack IA: A new stealthy spyware against voice assistance functions in smart phones," *CoRR*, vol. abs/1805.06187, pp. 1–11, May 2018.

[12] P. L. D. Leon, M. Pucher, J. Yamagishi, I. Hernaez, and I. Saratxaga, "Evaluation of speaker verification security and detection of HMM-based synthetic speech," *IEEE Trans. Audio, Speech, Language Process.*, vol. 20, no. 8, pp. 2280–2290, Oct. 2012.

[13] P. Cheng, I. E. Bagci, J. Yan, and U. Roedig, "Towards reactive acoustic jamming for personal voice assistants," in *Proc. 2nd Int. Workshop Multimedia Privacy Secur. (MPS)*, New York, NY, USA, Jan. 2018, pp. 12–17, doi: 10.1145/3267357.3267359.

[14] D. Kumar *et al.*, "Skill squatting attacks on Amazon Alexa," in *Proc. 27th USENIX Secur. Symp. (USENIX Security)*. Baltimore, MD, USA: USENIX Association, Aug. 2018, pp. 33–47. [Online]. Available: https://www.usenix.org/conference/usenixsecurity18/presentation/kumar

[15] P. Cheng, "Acoustic-channel attack and defence methods for personal voice assistants," Ph.D. dissertation, School Comput. Commun., Lancaster Univ., Lancaster, U.K., 2020.

[16] W. Stallings and L. Brown, *Computer Security: Principles and Practice*, 4th ed. London, U.K.: Pearson, 2018.

[17] *Australia Smart Speaker Consumer Adoption Report 2019*, Voicebot.ai, USA, Mar. 2019.

[18] D. Watkins, "Smart speakers and screens—Global smart speaker vendor & OS shipment and installed base market share by region: Q4 2018," Strategy Anal., Boston, MA, USA, Tech. Rep., Feb. 2019.

[19] Bloomberg. (Apr. 10, 2019). *Is Anyone Listening to You on Alexa? A Global Team Reviews Audio*. Accessed: Aug. 3, 2019. [Online]. Available: https://www.bloomberg.com/news/articles/2019-04-10/is-anyone-listening-to-you-on-alexa-a-global-team-reviews-audio

[20] The Verge. (Jul. 11, 2019). *Yep, Human Workers are Listening to Recordings From Google Assistant, Too*. Accessed: Aug. 3, 2019. [Online]. Available: https://www.theverge.com/2019/7/11/20690020/google-assistant-home-human-contractors-listening-recordings-vrt-nws

[21] BBC News. (Jul. 3, 2018). *Gavin Williamson Interrupted by Siri During Commons Statement*. Accessed: Aug. 3, 2019. [Online]. Available: https://www.bbc.com/news/av/uk-politics-44701007/gavin-williamson-interrupted-by-siri-during-commons-statement

[22] B. Karmann and T. Knudsen. (Jun. 30, 2019). *Project Alias*. Accessed: Aug. 5, 2019. [Online]. Available: https://github.com/bjoernkarmann/project_alias

[23] D. Schweppe. (Apr. 10, 2019). *Mycroft—Open Source Voice Assistant*. Accessed: Aug. 5, 2019. [Online]. Available: https://mycroft.ai

[24] European Commission. *2018 Reform of EU Data Protection Rules*. [Online]. Available: https://ec.europa.eu/commission/sites/beta-political/files/data-protection-factsheet-changes_en.pdf

[25] Office of the Attorney General. (2018). *California Consumer Privacy Act (CCPA)*. Accessed: Apr. 23, 2021. [Online]. Available: https://oag.ca.gov/privacy/ccpa

[26] W. Dienste and D. Bundestag, "Zulässigkeit der transkribierung und auswertung von mitschnitten der sprachsoftware 'Alexa' durch Amazon," Wissenschaftliche Dienste, Deutscher Bundestag, Germany, Tech. Rep. WD 10-3000-032/19, May 2019.

[27] *Information Privacy: Other Connected Device With a Voice Recognition Feature AB-1395*, California Legislature, CA, USA, Apr. 2019.

[28] N. Abdi, K. M. Ramokapane, and J. M. Such, "More than smart speakers: Security and privacy perceptions of smart home personal assistants," in *Proc. 15th Symp. Usable Privacy Secur. (SOUPS)*. Santa Clara, CA, USA: USENIX Association, Aug. 2019, pp. 451–466. [Online]. Available: https://www.usenix.org/conference/soups2019/presentation/abdi

[29] N. Roy, H. Hassanieh, and R. R. Choudhury, "BackDoor: Making microphones hear inaudible sounds," in *Proc. 15th Annu. Int. Conf. Mobile Syst., Appl., Services (MobiSys)*, New York, NY, USA, Jun. 2017, pp. 2–14, doi: 10.1145/3081333.3081366.

[30] D. Yu and L. Deng, *Automatic Speech Recognition: A Deep Learning Approach*. London, U.K.: Springer, 2014, doi: 10.1007/978-1-4471-5779-3.

[31] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, no. 2, pp. 257–286, Feb. 1989.

[32] G. E. Dahl, D. Yu, L. Deng, and A. Acero, "Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition," *IEEE Trans. Audio, Speech, Language Process.*, vol. 20, no. 1, pp. 30–42, Jan. 2012.

[33] H. Sak, A. W. Senior, K. Rao, and F. Beaufays, "Fast and accurate recurrent neural network acoustic models for speech recognition," *CoRR*, vol. abs/1507.06947, pp. 1–5, Jul. 2015.

[34] Google AI Blog. (2019). *An All-Neural On-Device Speech Recognizer*. Accessed: Oct. 10, 2019. [Online]. Available: https://ai.googleblog.com/2019/03/an-all-neural-on-device-speech.html

[35] D. Iter, J. Huang, and M. Jermann, "Generating adversarial examples for speech recognition," Stanford Univ., Stanford, CA, USA, Tech. Rep., 2017.

[36] N. Carlini and D. Wagner, "Audio adversarial examples: Targeted attacks on speech-to-text," in *Proc. IEEE Secur. Privacy Workshops (SPW)*, May 2018, pp. 1–7.

[37] X. Yuan *et al.*, "CommanderSong: A systematic approach for practical adversarial voice recognition," in *Proc. 27th USENIX Secur. Symp. (USENIX Security)*. Baltimore, MD, USA: USENIX Association, Aug. 2018, pp. 49–64. [Online]. Available: https://www.usenix.org/conference/usenixsecurity18/presentation/yuan-xuejing

[38] L. Zhang, Y. Meng, J. Yu, C. Xiang, B. Falk, and H. Zhu, "Voiceprint mimicry attack towards speaker verification system in smart home," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*. IEEE, 2020, pp. 377–386, doi: 10.1109/INFOCOM41043.2020.9155483.

[39] R. W. Shirey, *Internet Security Glossary, Version 2*, document RFC 4949, Aug. 2007. [Online]. Available: https://rfc-editor.org/rfc/rfc4949.txt

[40] S. Samonas and D. Coss, "The CIA strikes back: Redefining confidentiality, integrity and availability in security," *J. Inf. Syst. Secur.*, vol. 10, no. 3, pp. 1–25, 2014.

[41] C. Birge, "Enhancing research into usable privacy and security," in *Proc. 27th ACM Int. Conf. Design Commun. (SIGDOC)*. New York, NY, USA: Association for Computing Machinery, 2009, pp. 221–226, doi: 10.1145/1621995.1622039.

[42] J. Lau, B. Zimmerman, and F. Schaub, "'Alexa, stop recording': Mismatches between smart speaker privacy controls and user needs," in *Proc. 14th Symp. Usable Privacy Secur. (SOUPS)*, 2018, pp. 1–6.

[43] K. Bonilla and A. Martin-Hammond, "Older adults' perceptions of intelligent voice assistant privacy, transparency, and online privacy guidelines," in *Proc. 16th Symp. Usable Privacy Secur. (SOUPS)*, 2020, pp. 1–5.

[44] R. H. Taylor, A. Menciassi, G. Fichtinger, P. Fiorini, and P. Dario, "Medical robotics and computer-integrated surgery," in *Springer Handbook of Robotic*. Cham, Switzerland: Springer, 2016, pp. 1657–1684, doi: 10.1007/978-3-319-32552-1_63.

[45] N. Carlini *et al.*, "Hidden voice commands," in *Proc. 25th USENIX Secur. Symp. (USENIX Security)*. Austin, TX, USA: USENIX Association, Aug. 2016, pp. 513–530. [Online]. Available: https://www.usenix.org/conference/usenixsecurity16/technical-sessions/presentation/carlini

[46] P. Cheng, I. E. Bagci, J. Yan, and U. Roedig, "Smart speaker privacy control—Acoustic tagging for personal voice assistants," in *Proc. IEEE Secur. Privacy Workshops (SPW)*, May 2019, pp. 144–149.

[47] L. Zhang, S. Tan, and J. Yang, "Hearing your voice is not enough: An articulatory gesture based liveness detection for voice authentication," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur. (CCS)*, New York, NY, USA, Oct. 2017, pp. 57–71, doi: 10.1145/3133956.3133962.

[48] R. Naika, "An overview of automatic speaker verification system," in *Intelligent Computing and Information and Communication*, S. Bhalla, V. Bhateja, A. A. Chandavale, A. S. Hiwale, and S. C. Satapathy, Eds. Singapore: Springer, 2018, pp. 603–610.

[49] The Verge. (Oct. 11, 2017). *Amazon's Alexa Can Now Recognize Different Voices and Give Personalized Responses*. Accessed: Jun. 25, 2020. [Online]. Available: https://www.theverge.com/circuitbreaker/2017/10/11/16460120/amazon-echo-multi-user-voice-new-feature

[50] Google Assistant Help. *Link Your Voice to Your Google Assistant Device With Voice Match*. Accessed: Jun. 25, 2020. [Online]. Available: https://support.google.com/assistant/answer/9071681

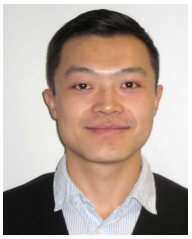
[51] X. Lei, G. Tu, A. X. Liu, C. Li, and T. Xie, "The insecurity of home digital voice assistants—Amazon Alexa as a case study," *CoRR*, vol. abs/1712.03327, pp. 1–12, Dec. 2017.

[52] J. Lindberg and M. Blomberg, "Vulnerability in speaker verification—A study of technical impostor techniques," in *Proc. 6th Eur. Conf.*

Speech Commun. Technol., 1999, pp. 1–4.
[53] F. Alegre, R. Vipperla, N. Evans, and B. Fauve, "On the vulnerability of automatic speaker recognition to spoofing attacks with artificial signals," in *Proc. 20th Eur. Signal Process. Conf. (EUSIPCO)*, Aug. 2012, pp. 36–40.
[54] Y. Stylianou, "Voice transformation: A survey," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Apr. 2009, pp. 3585–3588.
[55] Z. Wu and H. Li, "Voice conversion and spoofing attack on speaker verification systems," in *Proc. Asia–Pacific Signal Inf. Process. Assoc. Annu. Summit Conf.*, Oct. 2013, pp. 1–9.
[56] F. Kreuk, Y. Adi, M. Cisse, and J. Keshet, "Fooling end-to-end speaker verification with adversarial examples," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2018, pp. 1962–1966.
[57] M. Malekesmaeili and R. K. Ward, "A local fingerprinting approach for audio copy detection," *Signal Process.*, vol. 98, pp. 308–321, May 2014, doi: 10.1016/j.sigpro.2013.11.023.
[58] Q. Li, B.-H. Juang, and C.-H. Lee, "Automatic verbal information verification for user authentication," *IEEE Trans. Speech Audio Process.*, vol. 8, no. 5, pp. 585–596, Sep. 2000.
[59] T. Kinnunen *et al.*, "Utterance verification for text-dependent speaker recognition: A comparative assessment using the RedDots corpus," in *Proc. Interspeech*, 2016, pp. 430–434, doi: 10.21437/Interspeech.2016-1125.
[60] Z. Wu, N. Evans, T. Kinnunen, J. Yamagishi, F. Alegre, and H. Li, "Spoofing and countermeasures for speaker verification: A survey," *Speech Commun.*, vol. 66, pp. 130–153, Feb. 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167639314000788
[61] Z. Wu *et al.*, "ASVspoof 2015: The first automatic speaker verification spoofing and countermeasures challenge," in *Proc. 16th Annu. Conf. Int. Speech Commun. Assoc.*, Sep. 2015, pp. 1–5.
[62] Z. Wu *et al.*, "ASVspoof: The automatic speaker verification spoofing and countermeasures challenge," *IEEE J. Sel. Topics Signal Process.*, vol. 11, no. 4, pp. 588–604, Jun. 2017.
[63] H. Delgado *et al.*, "ASVspoof 2017 version 2.0: Meta-data analysis and baseline enhancements," in *Proc. Odyssey, Speaker Lang. Recognit. Workshop*, 2018.
[64] ASVspoof Consortium. (2019). *ASVspoof 2019: Automatic Speaker Verification Spoofing and Countermeasures Challenge Evaluation Plan*. Accessed: Jul. 28, 2020. [Online]. Available: https://www.asvspoof.org/asvspoof2019/
[65] J. Yamagishi *et al.*, "ASVspoof 2021: Accelerating progress in spoofed and deepfake speech detection," 2021, *arXiv:2109.00537*.
[66] S. Chen *et al.*, "You can hear but you cannot steal: Defending against voice impersonation attacks on smartphones," in *Proc. 37th IEEE Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Jun. 2017, pp. 183–195.
[67] H. Feng, K. Fawaz, and K. G. Shin, "Continuous authentication for voice assistants," in *Proc. 23rd Annu. Int. Conf. Mobile Comput. Netw. (MobiCom)*, New York, NY, USA, Oct. 2017, pp. 343–355, doi: 10.1145/3117811.3117823.
[68] C. Wang, S. A. Anand, J. Liu, P. Walker, Y. Chen, and N. Saxena, "Defeating hidden audio channel attacks on voice assistants via audio-induced surface vibrations," in *Proc. 35th Annu. Comput. Secur. Appl. Conf. (ACSAC)*, New York, NY, USA, 2019, pp. 42–56, doi: 10.1145/3359789.3359830.
[69] S. Pradhan, W. Sun, G. Baig, and L. Qiu, "Combating replay attacks against voice assistants," *Proc. ACM Interact. Mobile Wearable Ubiquitous Technol.*, vol. 3, no. 3, pp. 100:1–100:26, Sep. 2019, doi: 10.1145/3351258.
[70] L. Zhang, S. Tan, J. Yang, and Y. Chen, "VoiceLive: A phoneme localization based liveness detection for voice authentication on smartphones," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur. (CCS)*, New York, NY, USA, Oct. 2016, pp. 1080–1091, doi: 10.1145/2976749.2978296.
[71] L. Blue, L. Vargas, and P. Traynor, "Hello, is it me you're looking for? Differentiating between human and electronic speakers for voice interface security," in *Proc. 11th ACM Conf. Secur. Privacy Wireless Mobile Netw. (WiSec)*, New York, NY, USA, Jun. 2018, pp. 123–133, doi: 10.1145/3212480.3212505.
[72] Y. Lee *et al.*, "Using sonar for liveness detection to protect smart speakers against remote attackers," *Proc. ACM Interact., Mobile, Wearable Ubiquitous Technol.*, vol. 4, no. 1, pp. 1–28, Mar. 2020, doi: 10.1145/3380991.
[73] N. Roy, S. Shen, H. Hassanieh, and R. R. Choudhury, "Inaudible voice commands: The long-range attack and defense," in *Proc. 15th USENIX Symp. Netw. Syst. Design Implement. (NSDI)*. Renton, WA, USA: USENIX Association, Apr. 2018, pp. 547–560. [Online]. Available: https://www.usenix.org/conference/nsdi18/presentation/roy
[74] Y. He *et al.*, "Canceling inaudible voice commands against voice control systems," in *Proc. 25th Annu. Int. Conf. Mobile Comput. Netw. (MobiCom)*. New York, NY, USA: Association for Computing Machinery, Oct. 2019, pp. 1–15, doi: 10.1145/3300061.3345429.
[75] T. Vaidya, Y. Zhang, M. Sherr, and C. Shields, "Cocaine noodles: Exploiting the gap between human and machine speech recognition," in *Proc. 9th USENIX Workshop Offensive Technol. (WOOT)*. Washington, DC, USA: USENIX Association, Aug. 2015, pp. 1–14. [Online]. Available: https://www.usenix.org/conference/woot15/workshop-program/presentation/vaidya
[76] H. Abdullah, W. Garcia, C. Peeters, P. Traynor, K. R. B. Butler, and J. Wilson, "Practical hidden voice attacks against speech and speaker recognition systems," in *Proc. Netw. Distrib. Syst. Secur. Symp. (NDSS)*, 2019, pp. 1–15.
[77] H. Abdullah *et al.*, "Hear 'no evil', see 'kenansville'*: Efficient and transferable black-box attacks on speech recognition and voice identification systems," in *Proc. IEEE Symp. Secur. Privacy (SP)*, May 2021, pp. 712–729.
[78] M. Alzantot, B. Balaji, and M. B. Srivastava, "Did you hear that? Adversarial examples against automatic speech recognition," *CoRR*, vol. abs/1801.00554, pp. 1–16, Jan. 2018.
[79] R. Taori, A. Kamsetty, B. Chu, and N. Vemuri, "Targeted adversarial examples for black box audio systems," *CoRR*, vol. abs/1805.07820, pp. 1–6, Aug. 2018.
[80] S. Khare, R. Aralikatte, and S. Mani, "Adversarial black-box attacks on automatic speech recognition systems using multi-objective evolutionary optimization," in *Proc. Interspeech*, Sep. 2019, pp. 3208–3212.
[81] H. Yakura and J. Sakuma, "Robust audio adversarial example for a physical attack," *CoRR*, vol. abs/1810.11793, pp. 1–8, Oct. 2018.
[82] Y. Qin, N. Carlini, G. Cottrell, I. Goodfellow, and C. Raffel, "Imperceptible, robust, and targeted adversarial examples for automatic speech recognition," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 5231–5240.
[83] J. Szurley and J. Z. Kolter, "Perceptual based adversarial audio attacks," 2019, *arXiv:1906.06355*.
[84] L. Schönherr, T. Eisenhofer, S. Zeiler, T. Holz, and D. Kolossa, "Imperio: Robust over-the-air adversarial examples for automatic speech recognition systems," in *Proc. Annu. Comput. Secur. Appl. Conf. (ACSAC)*. New York, NY, USA: Association for Computing Machinery, Dec. 2020, pp. 843–855, doi: 10.1145/3427228.3427276.
[85] Z. Yang, B. Li, P.-Y. Chen, and D. Song, "Characterizing audio adversarial examples using temporal dependency," in *Proc. Int. Conf. Learn. Represent.*, 2019, pp. 1–15. [Online]. Available: https://openreview.net/forum?id=r1g4E3C9t7
[86] T. Chen, L. Shangguan, Z. Li, and K. Jamieson, "Metamorph: Injecting inaudible commands into over-the-air voice controlled systems," in *Proc. Netw. Distrib. Syst. Secur. Symp. (NDSS)*, 2020, pp. 1–17.
[87] Z. Li, Y. Wu, J. Liu, Y. Chen, and B. Yuan, "AdvPulse: Universal, synchronization-free, and targeted audio adversarial attacks via subsecond perturbations," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.* New York, NY, USA: Association for Computing Machinery, Oct. 2020, pp. 1121–1134, doi: 10.1145/3372297.3423348.
[88] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in *Proc. 3rd Int. Conf. Learn. Represent. (ICLR)*, Y. Bengio and Y. LeCun, Eds. San Diego, CA, USA, May 2015. [Online]. Available: http://arxiv.org/abs/1412.6572
[89] A. Y. Hannun *et al.*, "Deep speech: Scaling up end-to-end speech recognition," *CoRR*, vol. abs/1412.5567, pp. 1–12, Dec. 2014.
[90] N. Schinkel-Bielefeld, N. Lotze, and F. Nagel, "Audio quality evaluation by experienced and inexperienced listeners," in *Proc. Meetings Acoust. (ICA)*. Melville, NY, USA: Acoustical Society of America, 2013, Art. no. 060016.
[91] G. Zhang, X. Ji, X. Li, G. Qu, and W. Xu, "EarArray: Defending against DolphinAttack via acoustic attenuation," in *Proc. 28th Annu. Netw. Distrib. Syst. Secur. Symp.* Reston, VA, USA: The Internet Society, 2021, pp. 1–14. [Online]. Available: https://www.ndss-symposium.org/ndss-paper/eararray-defending-against-dolphinattack-via-acoustic-attenuation/
[92] Y. Chen *et al.*, "Devil's whisper: A general approach for physical adversarial attacks against commercial black-box speech recognition devices," in *Proc. 29th USENIX Secur. Symp. (USENIX Security)*. Berkeley, CA, USA: USENIX Association, Aug. 2020, pp. 2667–2684. [Online]. Available: https://www.usenix.org/conference/usenixsecurity20/presentation/chen-yuxuan
[93] J. Vadillo and R. Santana, "On the human evaluation of audio adversarial examples," 2020, *arXiv:2001.08444*.
[94] N. Zhang, X. Mi, X. Feng, X. Wang, Y. Tian, and F. Qian, "Dangerous skills: Understanding and mitigating security risks of voice-controlled third-party functions on virtual personal assistant systems," in *Proc. IEEE Symp. Secur. Privacy (SP)*, May 2019, pp. 1381–1396.
[95] C. Gao, V. Chandrasekaran, K. Fawaz, and S. Banerjee, "Traversing the quagmire that is privacy in your smart home," in *Proc. Workshop IoT Secur. Privacy (IoT S&P)*. New York, NY, USA: Association for Computing Machinery, Aug. 2018, pp. 22–28, doi: 10.1145/3229565.3229573.
[96] K. Sun, C. Chen, and X. Zhang, "Alexa, stop spying on me!': Speech privacy protection against voice assistants," in *Proc. 18th Conf. Embedded Netw. Sensor Syst. (SenSys)*. New York, NY, USA: Association for Computing Machinery, Nov. 2020, pp. 298–311, doi: 10.1145/3384419.3430727.
[97] (2021). *Raspberry Pi*. [Online]. Available: https://www.raspberrypi.org
[98] Y. He *et al.*, "Streaming end-to-end speech recognition for mobile devices," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 6381–6385.
[99] F. Brasser, T. Frassetto, K. Riedhammer, A.-R. Sadeghi, T. Schneider, and C. Weinert, "VoiceGuard: Secure and private speech processing," in *Proc. Interspeech*, Sep. 2018, pp. 1303–1307, doi: 10.21437/Interspeech.2018-2032.
[100] N. Tomashenko *et al.*, "Introducing the VoicePrivacy initiative," in *Proc. Interspeech*, 2020, pp. 169–1693, doi: 10.21437/Interspeech.2020-1333.
[101] F. Gontier, M. Lagrange, C. Lavandier, and J.-F. Petiot, "Privacy aware acoustic scene synthesis using deep spectral feature inversion," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2020, pp. 886–890.
[102] S.-X. Zhang, Y. Gong, and D. Yu, "Encrypted speech recognition using deep polynomial networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 5691–5695.
[103] A. Nautsch *et al.*, "Preserving privacy in speaker and speech characterisation," *Comput. Speech Lang.*, vol. 58, pp. 441–480, Nov. 2019.
[104] A. Mtibaa, D. Petrovska-Delacretaz, and A. B. Hamida, "Cancelable speaker verification

system based on binary Gaussian mixtures," in *Proc. 4th Int. Conf. Adv. Technol. Signal Image Process. (ATSIP)*, Mar. 2018, pp. 1–6.
[105] R. Gilad-Bachrach, N. Dowlin, K. Laine, K. Lauter, M. Naehrig, and J. Wernsing, "CryptoNets: Applying neural networks to encrypted data with high throughput and accuracy," in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 201–210.
[106] M. S. Riazi, M. Samragh, H. Chen, K. Laine, K. Lauter, and F. Koushanfar, "XONN: XNOR-based oblivious deep neural network inference," in *Proc. SEC*. Berkeley, CA, USA: USENIX Association, 2019, pp. 1501–1518.
[107] M. Dias, A. Abad, and I. Trancoso, "Exploring hashing and cryptonet based approaches for privacy-preserving speech emotion recognition," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2018, pp. 2057–2061.
[108] F. Teixeira, A. Abad, and I. Trancoso, "Patient privacy in paralinguistic tasks," in *Proc. 19th Annu. Conf. Int. Speech Commun. Assoc. (Interspeech)*, B. Yegnanarayana, Ed. Hyderabad, India: ISCA, Sep. 2018, pp. 3428–3432, doi: 10.21437/Interspeech.2018-2186.
[109] D. Leroy, A. Coucke, T. Lavril, T. Gisselbrecht, and J. Dureau, "Federated learning for keyword spotting," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 6341–6345.
[110] J. Geiping, H. Bauermeister, H. Dröge, and M. Moeller, "Inverting gradients—How easy is it to break privacy in federated learning?" in *Advances in Neural Information Processing Systems*, vol. 33, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, Eds. Red Hook, NY, USA: Curran Associates, 2020, pp. 16937–16947.
[111] F. Granqvist, M. Seigel, R. van Dalen, Á. Cahill, S. Shum, and M. Paulik, "Improving on-device speaker verification using federated learning with privacy," 2020, *arXiv:2008.02651*.
[112] J. Qian, H. Du, J. Hou, L. Chen, T. Jung, and X.-Y. Li, "Hidebehind: Enjoy voice input with voiceprint unclonability and anonymity," in *Proc. 16th ACM Conf. Embedded Netw. Sensor Syst. (SenSys)*, New York, NY, USA, Nov. 2018, pp. 82–94.
[113] Y. Gong and C. Poellabauer, "Crafting adversarial examples for speech paralinguistics applications," *CoRR*, vol. abs/1711.03280, pp. 1–8, Nov. 2017.
[114] A. Nelus and R. Martin, "Privacy-aware feature extraction for gender discrimination versus speaker identification," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 671–674.
[115] R. Aloufi, H. Haddadi, and D. Boyle, "Emotionless: Privacy-preserving speech analysis for voice assistants," 2019, *arXiv:1908.03632*.
[116] C. Champion, I. Olade, K. Papangelis, H. Liang, and C. Fleming, "The smart^2 speaker blocker: An open-source privacy filter for connected home speakers," *CoRR*, vol. abs/1901.04879, pp. 1–9, Jul. 2019.
[117] S. De Conca, "Between a rock and a hard place: Owners of smart speakers and joint control," *SCRIPTed*, vol. 17, p. 238, 2020.
[118] European Data Protection Supervisor and European Commission. *Techdispatch #1: Smart Speakers and Virtual Assistants*. Accessed: Nov. 13, 2021. [Online]. Available: https://edps.europa.eu/data-protection/our-work/publications/techdispatch/techdispatch-1-smart-speakers-and-virtual_en
[119] S. Sigg, L. N. Nguyen, P. P. Zarazaga, and T. Backstrom, "Provable consent for voice user interfaces," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun. Workshops (PerCom Workshops)*, Mar. 2020, pp. 1–4.
[120] R. Schlegel, K. Zhang, X. Zhou, M. Intwala, A. Kapadia, and X. Wang, "Soundcomber: A stealthy and context-aware sound trojan for smartphones," in *Proc. NDSS*, vol. 11, 2011, pp. 17–33.
[121] S. Narain, A. Sanatinia, and G. Noubir, "Single-stroke language-agnostic keylogging using stereo-microphones and domain specific machine learning," in *Proc. WiSec*, 2014, pp. 201–212.
[122] J. Liu, Y. Wang, G. Kar, Y. Chen, J. Yang, and M. Gruteser, "Snooping keystrokes with mm-level audio ranging on a single phone," in *Proc. 21st Annu. Int. Conf. Mobile Comput. Netw. (MobiCom)*. New York, NY, USA: Association for Computing Machinery, Sep. 2015, pp. 142–154, doi: 10.1145/2789168.2790122.
[123] I. Shumailov, L. Simon, J. Yan, and R. Anderson, "Hearing your touch: A new acoustic side channel on smartphones," *CoRR*, vol. abs/1903.11137, pp. 1–23, Mar. 2019.
[124] S. Shen, D. Chen, Y.-L. Wei, Z. Yang, and R. R. Choudhury, "Voice localization using nearby wall reflections," in *Proc. 26th Annu. Int. Conf. Mobile Comput. Netw. (MobiCom)*. New York, NY, USA: Association for Computing Machinery, Apr. 2020, pp. 1–14, doi: 10.1145/3372224.3380884.
[125] G. Diapoulis, C. Rosas, K. Larsson, and W. Kropp, "Person identification from walking sound on wooden floor," in *Proc. Euronoise*, 2018, pp. 1727–1732.
[126] D. Genkin, A. Shamir, and E. Tromer, "RSA key extraction via low-bandwidth acoustic cryptanalysis," in *Advances in Cryptology—CRYPTO 2014*, J. A. Garay and R. Gennaro, Eds. Berlin, Germany: Springer, 2014, pp. 444–461.
[127] D. Genkin, M. Pattani, R. Schuster, and E. Tromer, "Synesthesia: Detecting screen content via remote acoustic side channels," in *Proc. IEEE Symp. Secur. Privacy (SP)*, May 2019, pp. 853–869.
[128] M. Zhou *et al.*, "PatternListener: Cracking Android pattern lock using acoustic signals," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur. (CCS)*. New York, NY, USA: Association for Computing Machinery, Oct. 2018, pp. 1775–1787, doi: 10.1145/3243734.3243777.
[129] W. Wang, A. X. Liu, and K. Sun, "Device-free gesture tracking using acoustic signals," in *Proc. 22nd Annu. Int. Conf. Mobile Comput. Netw. (MobiCom)*. New York, NY, USA: Association for Computing Machinery, Oct. 2016, pp. 82–94, doi: 10.1145/2973750.2973764.
[130] R. Nandakumar, A. Takakuwa, T. Kohno, and S. Gollakota, "CovertBand: Activity information leakage using music," *Proc. ACM Interact., Mobile, Wearable Ubiquitous Technol.*, vol. 1, no. 3, pp. 87:1–87:24, Sep. 2017, doi: 10.1145/3131897.
[131] X. Xu, J. Yu, Y. Chen, Y. Zhu, L. Kong, and M. Li, "BreathListener: Fine-grained breathing monitoring in driving environments utilizing acoustic signals," in *Proc. 17th Annu. Int. Conf. Mobile Syst., Appl., Services (MobiSys)*. New York, NY, USA: Association for Computing Machinery, 2019, pp. 54–66, doi: 10.1145/3307334.3326074.

ABOUT THE AUTHORS



Peng Cheng received the B.E. degree in electronic science and technology from the Beijing University of Posts and Telecommunications, Beijing, China, in 2012, the M.E. degree in integrated circuit engineering from Tsinghua University, Beijing, the M.E. degree in electrical engineering from KU Leuven, Leuven, Belgium, in 2015, and the Ph.D. degree in computer science from Lancaster University, Lancaster, U.K., in 2020.

He was a Visiting Researcher with University College Cork, Cork, Ireland, from 2019 to 2020. He is currently a Postdoctoral Research Associate with the School of Cyber Science and Technology, Zhejiang University, Hangzhou, China. His current research interests include security and privacy related to acoustic channel and speech recognition systems in the context of the Internet of Things.

Utz Roedig received the Dr.Ing. degree in computer science from the Darmstadt University of Technology, Darmstadt, Germany, in 2002.

He joined the School of Computer Science and Information Technology (CSIT), University College Cork (UCC), Cork, Ireland, in January 2019. Since September 2021, he has been the Head of the School of Computer Science and Information Technology. Before moving to Cork, he was a Professor with Lancaster University, Lancaster, U.K., where he led the Academic Centre of Excellence in Cyber Security Research (ACE-CSR). Prior to his work at Lancaster University, he held research positions at UCC and the Darmstadt University of Technology. He is a Co-Principal Investigator of the Science Foundation Ireland (SFI)-funded CONNECT Centre for Future Networks and a Principal Investigator of the SFI Frontiers of the Future Award "Personal Voice Assistant Security and Privacy." Over the last number of years, his research has been supported by a number of research grants funded by EU, SFI, EPSRC, and industry. His research interests are computer networks and security with a focus on the Internet of Things (IoT). His work looks at the IoT communication mechanisms and the software used to construct the IoT systems with a particular focus on cybersecurity. He has published over 160 peer-reviewed articles in this field and his research collaborations with industry partners have resulted in several patents.

Dr. Roedig frequently serves as a TPC Member for international conferences, such as DCOSS, EWSN, and IPSN. He is a grant Reviewer for international funding bodies, such as EPSRC, U.K., ESF, EU, and FWO, Belgium.