The Feasibility of Injecting Inaudible Voice Commands to Voice Assistants

Chen Yan, Guoming Zhang, Xiaoyu Ji, *Member, IEEE,* Tianchen Zhang, Taimin Zhang and Wenyuan Xu, *Senior Member, IEEE,*

Abstract—Voice assistants (VAs) such as Siri and Google Now have become an increasingly popular human-machine interaction method and have made various systems voice controllable. Prior work on attacking voice assistants shows that the hidden voice commands that are incomprehensible to people can control the VAs. Hidden voice commands, though 'hidden', are nonetheless audible. In this work, we design a completely inaudible attack, `DolphinAttack`, that modulates voice commands on ultrasonic carriers to achieve inaudibility. By leveraging the nonlinearity of the microphone circuits, the modulated low-frequency audio commands can be successfully demodulated, recovered, and more importantly interpreted by the voice assistants. We validate `DolphinAttack` on popular voice assistants, including Siri, Google Now, S Voice, HiVoice, Cortana, Alexa, etc. By injecting a sequence of inaudible voice commands, we show a few proof-of-concept attacks, which include activating Siri to initiate a FaceTime call on iPhone, activating Google Now to turn on the airplane mode, and even manipulating the navigation system in an Audi automobile. We propose hardware and software defense solutions. We validate that it is feasible to detect `DolphinAttack` by classifying the audios using supported vector machine (SVM), and suggest to re-design voice assistants to be resilient to inaudible voice command attacks.

Index Terms—Voice Assistants, Speech Recognition, Microphones, Security Analysis, Defense.

1 Introduction

The recent technology advances in speech recognition have brought us closer to full-fledged artificial intelligent systems that can interact with human at the speed of interpersonal communication. Already, we have witnessed popular humanized 'voice assistants' (VAs) on a variety of systems: Apple Siri [1] and Google Now [2] on smartphones that allow users to initiate phone calls by voice, Alexa [3] on Amazon Echoes that enables users to place purchase orders, AI-powered voice assistant on Mercedes [4] that allows the driver to alter in-car settings handsfree. With the emerging of these voice assistants, it is important to understand how the voice assistants behave under intentional attacks.

Many security issues of voice assistants arise from the *difference* between how human and machines perceive voice. For voice assistants, the microphone hardware serves as the 'ear' that transforms acoustic waves to electrical signals, and the speech recognition software acts as the 'brain' that translates the signals into semantic information. Despite their decent functionality, the imperfect nature of hardware and software can open up chances for signals that are *unusual* in interpersonal communication to be accepted and correctly interpreted by voice assistants. This, however, enables sneaky attacks.

Prior studies [5], [6] focusing on the speech recognition software have shown that obfuscated voice commands which are incomprehensible to human can be understood by voice assistants. Such attacks, though 'hidden', are nonetheless audible and remain conspicuous. This paper analyzes the security of voice assistants from a hardware perspective, and aims at examining the feasibility of stealthier attacks that are otherwise impossible by manipulating the software. We are driven by the following key questions: *Can voice commands be* ***inaudible*** *to human while still being perceived and intelligible to voice assistants? Can injecting a sequence of inaudible voice commands lead to unnoticed security breaches to the entire system? To what extent can adversaries utilize the gap of human-machine difference?* To answer these questions, we designed `DolphinAttack`, an approach to inject inaudible voice commands at voice assistants by exploiting the ultrasound channel (i.e., $f > 20$ kHz) and the vulnerability of the underlying audio hardware.

Inaudible voice commands may appear to be unfeasible with the following doubts. (a) *How can inaudible sounds be audible to devices?* The upper bound frequency of human hearing is 20 kHz. Thus, most audio-capable devices (e.g., phones) adopt audio sample rates lower than 48 kHz, and apply low-pass filters to eliminate signals above 20 kHz [7]. Previous work [6] considers it impossible to receive voices above 20 kHz. (b) *How can inaudible sounds be intelligible to voice assistants?* Even if the ultrasound is received and correctly sampled by hardware, voice assistants will not recognize signals that do not match human tonal features, and are therefore unable to interpret commands. (c) *How can inaudible sounds be generated in a sneaky way?* Comparing with audible sounds, the generation of ultrasounds requires dedicated hardware and more transmitting power due to higher attenuation. Any attacks that are short-range or depend on equipment of significant size will be unpractical. We solved all these problems, and we show that the `DolphinAttack` voice commands, though totally imperceptible to human, can be received by the audio hardware of various devices, and correctly interpreted by voice assistants. We validated `DolphinAttack` on major voice assistants, including Siri, Google Now, Alexa, Samsung S Voice [8], Huawei HiVoice [9], Cortana [10], etc.

Furthermore, we characterize the security consequences by asking to what extent a sequence of inaudible voice commands can compromise the security of the system hosting voice assistants. We have tested DolphinAttack on 25 models of systems including Apple iPhone, Google Nexus, Amazon Echo, vehicles, etc. We believe the list is by far not comprehensive. Nevertheless, it serves as a wake-up call to reconsider what functionality and levels of human interaction shall be supported in voice assistants. To illustrate, we show that DolphinAttack can achieve the following sneaky attacks purely by a sequence of inaudible voice commands:

1) *Visiting a malicious website.* DolphinAttack voice commands can trick the device to open a malicious webpage, which can launch a drive-by-download attack or exploit a device with 0-day vulnerabilities.
2) *Spying.* An adversary can let the victim device start outgoing video/phone calls, therefore accessing the visual/acoustic surroundings of the device.
3) *Injecting fake information.* An adversary may make the victim device send fake messages or emails, add fake online posts, insert fake events to a calendar, etc.
4) *Denial of service.* An adversary may turn on the airplane mode, disconnecting all wireless communications.
5) *Concealing attacks.* The screen and voice feedback may expose the attacks. The adversary may decrease the odds by dimming the screen and lowering the volume.

DolphinAttack voice commands are made feasible because of the widely existed hardware vulnerabilities and challenge the common design assumption that adversaries may at most try to manipulate a voice assistant vocally and can be detected by an alert user. In addition, DolphinAttack does not require adversaries to be physically close to the victim devices. Adversaries can inject inaudible voice commands at nearly 20 m away with a transmitter array or exploit remotely accessible commodity speakers to attack devices nearby. To address these widely existed security issues, we generalize the method of DolphinAttack as a building block to study vulnerabilities and propose both hardware and software solutions.

In summary, we list our contributions as follows.

- We present DolphinAttack that can inaudibly inject voice commands at state-of-the-art voice assistants by exploiting ultrasounds and the vulnerabilities of audio hardware. We validate DolphinAttack on 12 popular voice assistants (Siri, Google Now, Alexa, etc.) across 25 models of devices (smartphones, speakers, cars, etc.).
- We show that adversaries can achieve a series of highly practical attacks by injecting a sequence of inaudible voice commands with either portable, long-range, or remote setups. Tested attacks include launching FaceTime on iPhones, shopping on an Amazon Echo, manipulating the navigation system in an Audi automobile, etc.
- We suggest both hardware and software based defense strategies to alleviate the attacks, and we provide suggestions to enhance the security of voice assistants.

2 Background and Threat Model

In this section, we introduce popular voice assistants, discuss their architecture with a focus on microphones, and propose the threat model.

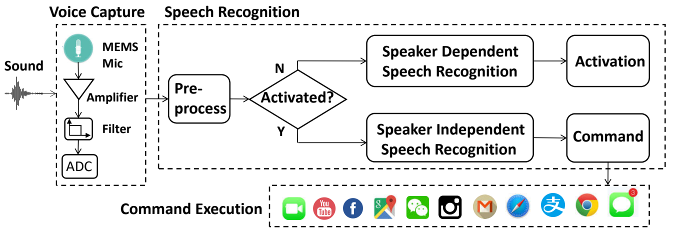


Fig. 1: The architecture of a state-of-the-art VA that can take voice commands as inputs and execute commands.

2.1 Voice Assistants

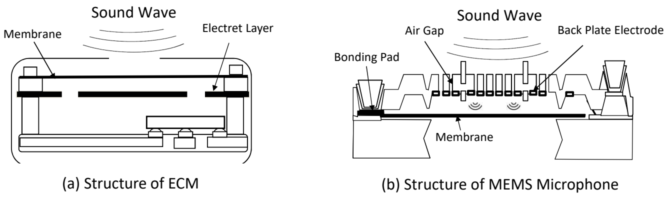
A typical voice assistant (VA) system consists of three main sub-systems: *voice capture*, *speech recognition*, and *command execution*, as illustrated in Fig. 1. The voice capture subsystem records ambient voices, which are amplified, filtered, and digitized, before being passed into the speech recognition subsystem. Then, the raw captured digital signals are first pre-processed to remove frequencies that are beyond the audible sound range and to discard signal segments that contain sounds too weak to be identified. Next, the processed signals enter the speech recognition system.

Typically, a speech recognition (SR) system works in two phases: activation and recognition. During the activation phase, the system cannot accept arbitrary voice inputs, but it waits to be activated. To activate the system, a user has to either say pre-defined wake words or press a special button. For instance, Amazon echo takes “Alexa” as the activation wake word. Apple Siri can be activated by pressing and holding the home button for about one second or by “Hey Siri” if the *'Allow Hey Siri'* feature is enabled. To recognize the wake words, the microphones continue recording ambient sounds until a voice is collected. Then, the systems will use either speaker-dependent or speaker-independent speech recognition algorithm to recognize the voice. For instance, the Amazon Echo exploits speaker-independent algorithms and accepts “Alexa” spoken by any one as long as the voice is clear and loud. In comparison, Apple Siri is speaker dependent. Siri requires to be trained by a user and only accepts “Hey Siri” from the same person. Once activated, the SR system enters the recognition phase and will typically use speaker-independent algorithms to convert voices into texts, i.e., commands in our cases.

Note that a speaker-dependent SR is typically performed locally and a speaker-independent SR is performed via a cloud service [11]. To use the cloud service, the processed signals are sent to the servers, which will extract features (typically Mel-frequency cepstral coefficients [12], [13]) and recognize commands via machine learning algorithms (e.g., the Hidden Markov Models or neural networks). Finally, the commands are sent back.

Given a recognized command, the command execution system will launch the corresponding application or execute an operation. The acceptable commands and corresponding actions are system dependent and defined beforehand. Popular voice assistants have been built on smartphones, wearable devices, smart home devices, and automobiles. Smartphones allow users to perform a wide range of operation via voice commands, such as dialing a phone number, sending short messages, opening a webpage, setting the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TDSC.2019.2906165, IEEE Transactions on Dependable and Secure Computing



(a) Structure of ECM

(b) Structure of MEMS Microphone

Fig. 2: An illustration of the electret condenser microphone (ECM) and MEMS microphone structure.

phone to the airplane mode, etc. Modern automobiles accept an elaborate set of voice commands to activate and control a few in-car features, such as navigation, the entertainment system, the environmental controls, and mobile phones. For instance, if "Call 1234567890" is recognized, an automobile or a smartphone may start dialing the phone number 1234567890.

Many security studies on voice assistants focus on attacking either the speech recognition algorithms [5] or command execution environment (e.g., malware). This paper aims at the voice capturing subsystem, which will be detailed in the next subsection.

2.2 Microphone

A voice capture subsystem records audible sounds mainly by a microphone, which is a transducer that converts airborne acoustic waves (i.e., sounds) to electrical signals. A majority of microphones are condenser microphones, and two types of condenser microphones are used on voice controllable devices: electret condenser microphones (ECMs) and microelectromechanical system (MEMS) microphones. Due to the miniature package sizes, lower power consumption and excellent temperature characteristics, MEMS microphones dominate mobile devices, including smartphones and wearables. Nevertheless, ECMs and MEMS microphones work similarly. As shown in Fig. 2, condenser microphones are air-gapped capacitors that contain a movable membrane and a fixed electrode (electret for ECMs) [14]. In the presence of a sound wave, the air pressure caused by the sound wave reaches the membrane, which flexes in response to changes in air pressure, while the other electrode remains stationary. The movement of the membrane creates a change in the amount of capacitance between the membrane and the fixed electrode. Since a nearly constant charge is maintained on the capacitor, the capacitance changes will produce an AC signal. In this way, air pressure is converted into an electrical signal.

Designed to capture audible sounds, microphones, low-pass filters (LPFs), and ADC in the voice capture subsystem are all designed to suppress signals out of the frequency range of audible sounds (i.e., 20 Hz to 20 kHz). According to datasheets, the sensitivity spectrum of microphones is between 20 Hz to 20 kHz. Ideally, even if a signal higher than 20 kHz is recorded by a microphone, it will be removed by the LPF. Finally, the sample rate of the ADC is typically 44.1 kHz, and the digitized signal's frequency is limited below 22 kHz according to the Nyquist Sampling Theorem.

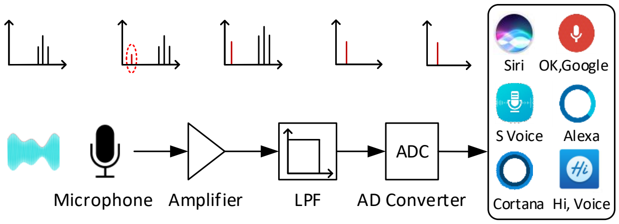


Fig. 3: An illustration on the modulated tone traversing the signal pathway of a voice capture device in terms of FFT.

2.3 Threat Model

The adversary's goal is to inject voice commands into a voice controllable device without the owners' awareness, and perform unauthorized actions. We assume that the adversary owns equipment that transmits acoustic signals, but she has no direct access to the targeted device and cannot force the owner to perform any tasks.

No Target Device Access. We assume that an adversary may target at any voice assistants of her choices, but she has no direct access to the target devices. She cannot physically touch them, alter the device settings, or install malware. However, we assume that she is fully aware of the characteristics of the target devices. Such knowledge can be gained by acquiring and analyzing devices of the same model beforehand.

No Owner Interaction. We assume that the target device may be in the owner's vicinity, but may not be in use and draw no attention (e.g., on the other side of a desk, with screen covered, or in a pocket). In addition, the device may be unattended, which can happen when the owner is temporarily away (e.g., leaving an Amazon Echo in a room). Nevertheless, the adversaries cannot ask the owners to perform any operations, such as pressing a button or unlocking the device.

Inaudible. Since the goal of an adversary is to inject voice commands without being detected, she will use the sounds inaudible to human, i.e., ultrasounds ($f > 20$ kHz). Note that we did not use near-ultrasonic sounds (18 kHz $< f <$ 20 kHz) because they are still audible to kids.

Attacking Equipment. We assume that an adversary could acquire either professional ultrasonic speakers or those designed for playing audible sounds. An attacking speaker is assumed to be in the vicinity of the target devices. For instance, the adversary may secretly exploit a remote controllable speaker around the victim's desk or home. Alternatively, she may also carry a portable speaker while walking by the victim.

3 Feasibility Analysis

The fundamental idea of DolphinAttack is (a) to modulate the low-frequency voice signal (i.e., baseband) on an ultrasonic carrier before transmitting it over the air, and (b) to demodulate the modulated voice signals with the voice capture hardware at the receiver. Since we have no control on the voice capture hardware, we have to craft the modulated signals in such a way that it can be demodulated to the baseband signal using the voice capture hardware as it is. Given that microphone modules always utilize LPF to

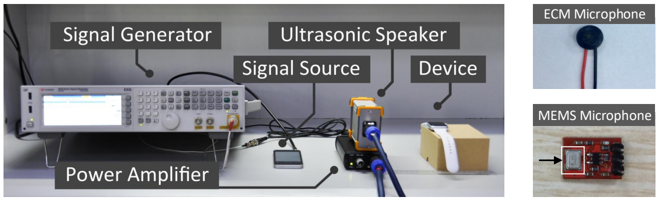


Fig. 4: An illustration of the benchtop experimental setup for investigating the feasibility of receiving ultrasounds with ECM and MEMS microphones. This benchtop setup is used for validating the feasibility of attacking various voice assistants as well.

suppress undesired high-frequency signals, the demodulation shall be accomplished prior to LPF.

Since the signal pathway of voice capture hardware starts from a microphone, one or more amplifiers, LPF, to ADC, the potential components for demodulation are microphones and amplifiers. We look into the principle of both to accomplish `DolphinAttack`. Although electric components such as amplifiers are designed to be linear, in reality they exhibit **nonlinearity**. With this nonlinearity property, the electric component is able to create new frequencies [15]. Although the nonlinearity of amplifier modules has been reported and utilized, it remains unknown whether microphones, including both the MEMS microphones and the ECMs possess such a property.

To investigate, we first theoretically model the nonlinearity of a microphone module, and then validate the nonlinearity effect on real microphone modules.

3.1 Nonlinearity Effect Modeling

A microphone converts mechanical sound waves into electrical signals. Essentially, a microphone can be roughly considered as a component with square-law non-linearity in the input/output signal transfer characteristics. Amplifiers are known to have nonlinearity, which can produce demodulated signals in the low-frequency range [16]. In this paper, we study the nonlinearity of microphones and we can model it as the following. Let the input signal be $s_{in}(t)$, and the output signal $s_{out}(t)$ be

$$s_{out}(t) = As_{in}(t) + Bs_{in}^2(t) \tag{1}$$

where A is the gain for the input signal and B is the gain for the quadratic term s_{in}^2. A linear component takes a sinusoidal input signals of frequency f and outputs a sinusoidal signal with the same frequency f. In comparison, the nonlinearity of electronic devices can produce harmonics and cross-products[1], and enable the devices to generate new frequencies, i.e., with a crafted input signal they can down-convert the signal as well as recover the baseband signal.

Suppose the signal of a voice command is $m(t)$. We modulate the signal on an ultrasound carrier with central frequency f_c, and let the modulated signal be

$$s_{in}(t) = m(t)\cos(2\pi f_c t) + \cos(2\pi f_c t) \tag{2}$$

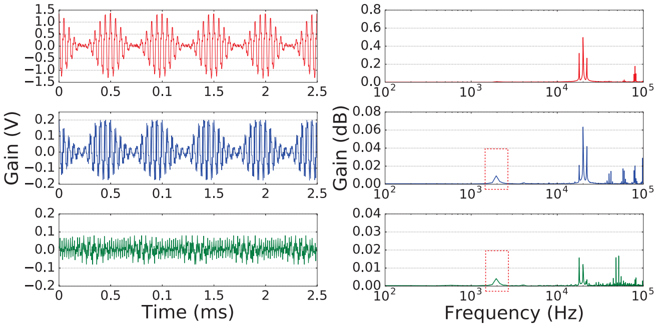


Fig. 5: Evaluation of the nonlinearity effect. The time and frequency domain plots for the original signal, the output signal of the MEMS microphone, and the output signal of the ECM microphone. The presence of baseband signals at 2 kHz shows that nonlinearity can demodulate the signals.

That is, amplitude modulation is used. Without loss of generality, let $m(t)$ be a single tone, i.e., $m(t) = \cos(2\pi f_m t)$. After applying Eq. (2) to Eq. (1) and taking the Fourier transform, we can confirm that the output signal contains the intended baseband frequency f_m together with the fundamental frequency components of s_{in} (i.e., $f_c - f_m$, $f_c + f_m$, and f_c), harmonics, and other cross products (i.e., $2f_m, 2(f_c - f_m), 2(f_c + f_m), 2f_c, 2f_c + f_m$, and $2f_c - f_m$). After a LPF, all high-frequency components will be removed and the f_m frequency component will remain, which completes the down-conversion, as shown in Fig. 3. Note that other unfiltered frequencies (e.g., $2f_m$) can bring distortion.

3.2 Nonlinearity Effect Evaluation

Given the theoretical calculation of the nonlinearity effect of the microphone module and its influence on the input signal after modulation, in this section, we verify the nonlinearity effect on real microphones. We test both types of microphones: ECM and MEMS microphones.

3.2.1 *Experimental Setup*

The experimental setup is shown in Fig. 4. We use an iPhone SE smartphone to generate a 2 kHz single-tone signal, i.e., the baseband signal. The baseband signal is then inputted to a vector signal generator [17], which modulates the baseband signal onto an ultrasonic carrier. After amplified by a power amplifier, the modulated signal is transmitted by a high-quality full-band ultrasonic speaker [18].

On the receiver side, we test an ECM extracted from a headset and an ADMP401 MEMS microphone [19]. As is shown in Fig. 4, the ADMP401 microphone module contains a preamplifier. To understand the characteristics of microphones, we measure the signal output from the microphone instead of from the preamplifier.

3.2.2 *Results*

We study the nonlinearity using two types of signals: single tones and voices with multiple tones.

Single Tone. Fig. 5 shows the result when we use a 20 kHz carrier, which confirms that the nonlinearity of the microphone manages to demodulate the baseband signals. The top two figures show the original signal from the

[1] Harmonics are frequencies that are integer multiples of the fundamental frequency components, and cross-products are multiplicative or conjunctive combinations of harmonics and fundamental frequency components.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TDSC.2019.2906165, IEEE Transactions on Dependable and Secure Computing

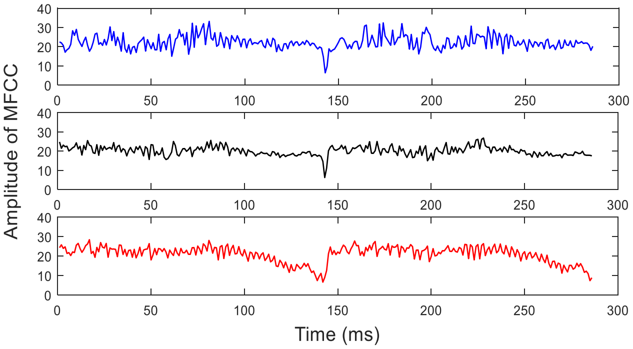


Fig. 6: The MFCC for three sound clips of “Hey”. From top to bottom: the TTS generated voice, the recorded voice as the TTS voice is played in audible sounds, the recorded voice as the TTS voice is modulated to 25 kHz.

speaker in the time domain and the frequency domain, whereby the carrier frequency (20 kHz) and an upper side band as well as a lower sideband (20 ± 2 kHz) appear nicely. The two figures in the second row show the output signal from the MEMS microphone and the bottom ones are from the ECM. Even though the signals were attenuated, especially for the ECM, the baseband (2 kHz) in the frequency domain for both microphones confirm the success of demodulation. We note that the MEMS microphones receive stronger signals at both 20 kHz and 2 kHz than the ECMs, which is possibly because the miniature size of MEMS microphones makes them more sensitive to sounds of shorter wavelengths, i.e., ultrasounds. Note that the frequency domain plots also include the harmonics higher than 20 kHz, and they will be filtered by the LPF and shall not affect the speech recognition.

Voices. Even though we can demodulate a signal tone successfully, voices are a mix of numerous tones at various frequencies, and it is unknown whether a demodulated voice signal remains similar to the original one. Thus, we calculated the Mel-Frequency Cepstral Coefficients (MFCC), one of the most widely used features of sounds, of three sound clips of “Hey”: (a) the original voice generated by a text-to-speech (TTS) engine, (b) the voice recorded by a Samsung Galaxy S6 Edge as an iPhone 6 plus plays the original TTS voice, and (c) the voice recorded by a Samsung S6 Edge as the TTS voices are modulated and played by the ultrasonic speaker Vifa. As Fig. 6 shows, the MFCC of all three cases are similar. To quantify the similarity, we calculate the Mel-Cepstral Distortion (MCD) between the original one and the recorded ones, which is 3.1 for case (b) and 7.6 for case (c). MCD quantifies the distortion between two MFCCs, and the smaller the better. Typically, the two voices are considered to be acceptable to voice recognition systems if their MCD values are smaller than 8 [20], and thus the result encourages us to carry out further study on `DolphinAttack` against voice assistants.

4 Attack Design

`DolphinAttack` utilizes inaudible voice injection to control VAs silently. Since attackers have little control of the VAs, the key of a successful attack is to generate inaudible voice commands at the attacking transmitter. In particular, `DolphinAttack` has to generate the baseband signals of voice commands for both activation and recognition phases of the VAs, modulate the baseband signals such that they can be demodulated at the VAs efficiently, and design a portable transmitter that can launch `DolphinAttack` anywhere. The basic building blocks of `DolphinAttack` are shown in Fig. 7, and we discuss these details in the following subsections. Without loss of generality, we use Siri as a case study, and the technology can be applied to other voice assistants (e.g., Google Now, Alexa) easily.

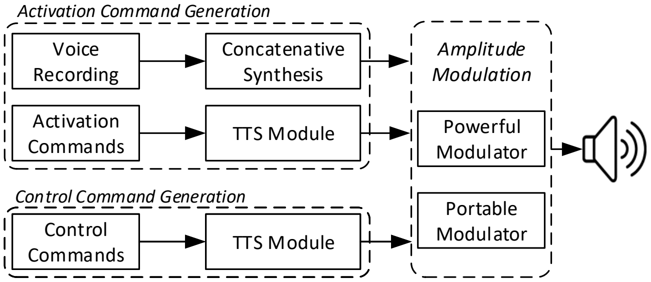


Fig. 7: Architecture of the transmitter modules. The transmitter mainly includes the command generation modules and the modulation module.

4.1 Voice Commands Generation

Siri works in two phases: activation and recognition. It requires activation before accepting voice commands, and thus we generate two types of voice commands: activation commands and general control commands. To control a VA, `DolphinAttack` has to generate activation commands before injecting general control commands.

4.1.1 Activation Commands Generation

A successful activation command has to satisfy two requirements: (a) containing the wake words “Hey Siri”, and (b) toning to the specific voice of the user that was trained for Siri. We design two methods to generate activation commands for two scenarios respectively: (a) an attacker cannot identify the owner of Siri (e.g., attacking random users), and (b) an attacker can obtain a few recordings of the owner’s voice.

(1) TTS-Based Brute Force. The recent advance in Text-to-Speech (TTS) technology makes it easy to convert texts to voices. The only challenge of generating activation commands with TTS is how to match the timbre of TTS voice with that of a human user. We observed that two users with similar vocal tones can activate the other’s Siri. Thus, as long as there is one TTS voice that is close enough to the owner’s, it suffices to activate Siri. In `DolphinAttack`, we prepare a set of activation commands in various tones and timbres with the help of existing TTS systems (summarized in Tab. 1), which include Selvy Speech, Baidu, Google, etc. In total, we obtain 90 types of TTS voices.

(2) Concatenative Synthesis. When an attacker can record a few recordings from the owner of the Siri which do not include “Hey Siri”, we propose to synthesize a desired voice command by searching for relevant phonemes (i.e., HH, EY, S, IH, R) from other words in the available recordings.

TABLE 1: The list of TTS systems used for attacking the Siri trained by the Google TTS [21], and the evaluation results on activation and control commands.

TTS Systems	Voice Type #	# of successful types	
		Call 12..90	*Hey Siri*
Selvy Speech [22]	4	4	2
Baidu [23]	1	1	0
Sestek [24]	7	7	2
NeoSpeech [25]	8	8	2
Innoetics [26]	12	12	7
Vocalware [27]	15	15	8
CereProc [28]	22	22	9
Acapela [29]	13	13	1
Fromtexttospeech [30]	7	7	4

4.1.2 General Control Commands Generation

General control commands can be any commands that launch applications (e.g., "call 911", "open google.com") or configure the devices (e.g., "turn on airplane mode"). Unlike the activation commands, SR systems generally do not authenticate the control commands. Thus, an attacker can choose the text of any control commands and utilize TTS systems to generate them.

4.1.3 Evaluation

We test both activation and control commands. Without loss of generality, we generate both activation and control commands with the TTS systems summarized in Tab. 1. In particular, we prepare two voice commands: "Hey Siri" and "Call 1234567890". For activation, we use the voices from the Google TTS system to train Siri, and the rest for testing. We play the voice commands with an iPhone 6 Plus and the benchtop devices in Fig. 4, and test on an iPhone 4S. The results of both activation and recognition commands are summarized in Tab. 1, which show that the control commands from all of the TTS systems can be recognized by the SR system, and 35 out of 89 types of activation commands can activate Siri, resulting in a success rate of 39 percent. The results are similar when Siri is trained by a human user.

4.2 Voice Commands Modulation

After generating the baseband voice commands, we need to modulate them on ultrasonic carriers such that they are inaudible. To leverage the nonlinearity of microphones, `DolphinAttack` has to utilize amplitude modulation (AM).

4.2.1 Amplitude Modulation Parameters

In AM, the amplitude of the carrier wave varies in proportion to the baseband signal, and amplitude modulation produces a signal with its power concentrated at the carrier frequency and two adjacent sidebands. In the following, we describe how to select AM parameters in `DolphinAttack`.

(1) Depth. Modulation depth m is defined as $m = M/A$, where A is the carrier amplitude, and M is the modulation amplitude, i.e., M is the peak change in the amplitude from its unmodulated value. For example, if $m = 0.5$, the carrier amplitude varies by 50 percent above (and below) its unmodulated level. Modulation depth is directly related to the utilization of the nonlinearity effect of microphones, and our experiments show that the modulation depth is hardware dependent (detailed in Sec. 5).

(2) Carrier Frequency. The selection of the carrier frequency depends on several factors: the frequency range of ultrasounds, the bandwidth of the baseband signal, the cut-off frequency of the low pass filter and the frequency response of the microphone, as well as the frequency response of the attacking speaker. The lowest frequency of the modulated signal should be larger than 20 kHz to ensure inaudibility. Let the frequency range of a voice command be w, the carrier frequency f_c has to satisfy the condition $f_c - w > 20$ kHz. For instance, given that the bandwidth of the baseband is 6 kHz, the carrier frequency has to be larger than 26 kHz to ensure that the lowest frequency is larger than 20 kHz. One may consider using carriers with frequencies right below 20 kHz, because sounds at these frequencies are still inaudible to most people except kids. However, such carriers will not be effective. This is because when the carrier frequency and lower sideband are below the cut-off frequency of the low-pass filter, they will not be filtered. Therefore, the recovered voices are different from the original signals, and the SR systems will fail to recognize the commands.

Similar to many electronic devices, microphones and speakers are frequency selective, e.g., the gains at different frequencies vary. For efficiency, the carrier frequency shall be the one that has the highest product of the gains at both the speaker and the microphone. To explore, we measure the frequency response of a few speakers and microphones, i.e., given the same stimulus, we measure the output magnitude at various frequencies. The results show that the gains of the microphones and speakers do not necessarily decrease with the increase of frequencies, thus the effective carrier frequencies may not be monotonous, and can be device-dependent.

(3) Voice Selection. Various voices map to various baseband frequency ranges. For example, a female voice typically has a wider frequency band than what a male voice has, which results in a larger probability of frequency leakage over audible frequency range, i.e., the lowest frequency of the modulated signal may be smaller than 20 kHz. Thus, if possible, a voice with a small bandwidth shall be selected to create baseband voice signals.

4.3 Voice Commands Transmitter

We design two transmitters: (a) a powerful transmitter that is driven by a dedicated signal generator (shown in Fig. 4) and (b) a portable transmitter that is driven by a smartphone (shown in Fig. 8). We utilize the first one to validate and quantify the extent to which `DolphinAttack` can accomplish various inaudible voice commands, and we use the second one to validate the feasibility of *a walk-by attack*. Both transmitters consist of three components: a signal source, a modulator, and a speaker. The signal source produces baseband signals of the original voice commands, and outputs to the modulator, which modulates the voice signal onto a carrier wave of frequency f_c in forms of amplitude modulation. Finally, the speaker transforms the modulated signals into acoustic waves.

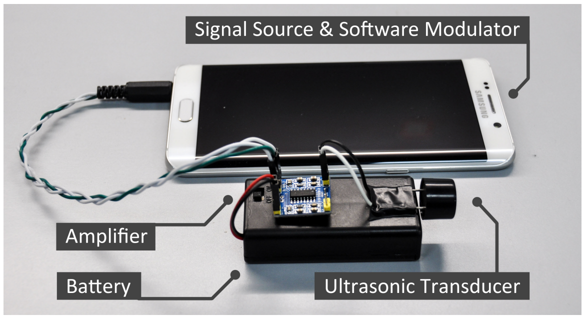


Fig. 8: A portable transmitter implemented with a Samsung Galaxy S6 edge+ smartphone, a low-cost amplifier and an ultrasonic transducer. The total cost of the amplifier, the ultrasonic transducer and the battery is less than $3.

4.3.1 The Powerful Transmitter with a Signal Generator

We utilize a smartphone as the signal source and the vector signal generator described in Fig. 4 as the modulator. Note that the signal generator has a sampling range of 300 MHz, much larger than ultrasonic frequencies, and can modulate signals with predefined parameters. The speaker of the powerful transmitter is a wide-band dynamic ultrasonic speaker named Vifa [18].

4.3.2 The Portable Transmitter with a Smartphone

The portable transmitter utilizes a smartphone to generate the modulated signals. Since we found that the best carrier frequencies for many devices are larger than 24 kHz as is depicted in Tab. 3, a majority of smartphones cannot accomplish the task. Most smartphones support at most a 48 kHz sample rate and can only transmit a modulated narrow-band signal with the carrier frequency of at most 24 kHz. To build a portable transmitter that works for a wide range of devices, we acquired a Samsung Galaxy S6 edge+, which supports a sample rate up to 192 kHz when using external speakers. We adopt a narrow-band ultrasonic transducer [31] as the speaker and power it with an amplifier as shown in Fig. 8. As such, the effective attacking distance is extended.

5 Feasibility Experiments across VAs

We validate `DolphinAttack` experimentally on 12 popular voice assistants and 25 models of devices (27 devices in total), and seek answers to three questions: (a) Will the attacks work against different voice assistants on various operation systems and hardware platforms? (b) How do various software and hardware affect the performance of attacks? (c) What are the key parameters in crafting a successful attack? This section describes the experiment design, setup, and results in detail.

5.1 System Selection

We examine `DolphinAttack` on various state-of-the-art voice assistants and off-the-shelf devices, which are categorized in Tab. 2 and listed in Tab. 3. The list does not intend to be exhaustive, but rather provides a representative set of voice assistants and devices that can be acquired for experiments with our best effort.

TABLE 2: The list of voice commands used in the experiment on the devices and systems in Tab. 3.

Attack	Device/Voice Assistant	Voice Command
Recognition	Smartphones & Wearable	*Call 1234567890*
	iPad	*FaceTime 1234567890*
	MacBook & Nexus 7	*Open dolphinattack.com*
	Windows PC	*Turn on airplane mode*
	Smart speakers	*Open the back door*
	Vehicles	*Navigation* *
Activation	Apple Siri	*Hey Siri*
	Google Now	*Ok Google*
	Samsung S Voice	*Hi Galaxy*
	Huawei HiVoice	*Hello Huawei* *
	Huawei HiAssistant	*Hello Xiaoyi* *
	Alexa	*Alexa*
	Cortana	*Hey Cortana*
	iFlyTek	*Dingdong Dingdong* *
	AliGenie	*Tmall Genie* *
	Banma	*Hello Banma* *

* Spoken in Chinese due to the lack of language support on English.

Our approach in selecting the target systems is twofold—software and hardware. First of all, we select major voice assistants that are publicly available, e.g., Siri, Google Now, Alexa, Cortana, etc. Unlike ordinary software, voice assistants (especially proprietary ones) are highly hardware and OS dependent. For example, Siri is limited Apple products and Cortana runs exclusively on Windows machines. Nevertheless, we select and experiment on the hardware whichever the voice assistants are compatible with. To explore the hardware influence on the attack performance, we also examine the attacks on different hardware models running the same voice assistant, e.g., Siri on various generations of iPhones.

In summary, we select voice controllable devices and voice assistants (shown in Tab. 3) that are popular on the consumer market with active users and cover various application areas as well as usage scenarios. They can be classified into four categories—personal devices (smartphones, tablets, wearables), computers, smart home devices (speakers), and vehicles.

5.2 Experiment Setup

We test `DolphinAttack` on each of the selected device and voice assistant with the same experiment setup, and report their behaviors under attack with three goals:

- Examining the feasibility of attacks.
- Quantifying the parameters in tuning a successful attack.
- Measuring the attack performance.

Equipment. Unless specified, all experiments utilize the default experiment equipment—the powerful transmitter shown in Fig. 4. Since the powerful transmitter is able to transmit signals with a wide range of carriers (from 9 kHz to 50 kHz), we use it for feasibility study. In comparison, the portable transmitter utilizes narrow-band speakers, and its transmission frequencies are limited by the available narrow-band speakers. In our case, our portable transmitter can transmit signals at the frequencies of 23 kHz, 25 kHz, 33 kHz, 40 kHz, and 48 kHz.

Setup. As shown in Fig. 4, we position a target device on a table in front of the ultrasonic speaker at varying distances, with the device microphone facing right toward

TABLE 3: Experimented devices, VAs, and the results. The examined attacks include *recognition* (after the voice assistants have been manually activated) and *activation* (when the voice assistants are not activated). The voice assistants are trained with the voice of Google TTS, and the same voice is used in recognition and activation attacks. The modulation parameters and maximum attack distances are measured in an office environment with a background noise of 55 dB SPL.

Type	Manuf.	Model	OS/Version	Voice Assistant	Attacks		Modulation Parameters		Max Dist. (cm)	
					Recog.	Activ.	f_c (kHz) & [Prime f_c] ‡	Depth	Recog.	Activ.
Smartphone & Tablet & Wearable	Apple	iPhone 4s	iOS 9.3.5	Siri	√	√	20–42 [27.9]	$\geq$ 9%	175	110
	Apple	iPhone 5s	iOS 10.0.2	Siri	√	√	24.1 26.2 27 29.3 [24.1]	100%	7.5	10
	Apple	iPhone SE	iOS 10.3.1	Siri	√	√	22–28 33 [22.6]	$\geq$ 47%	30	25
				Chrome	√	N/A	22–26 28 [22.6]	$\geq$ 37%	16	N/A
	Apple	iPhone SE †	iOS 10.3.2	Siri	√	√	21–29 31 33 [22.4]	$\geq$ 43%	21	24
	Apple	iPhone 6 Plus *	iOS 10.3.1	Siri	×	√	— [24]	—	—	2
	Apple	iPhone 6 Plus *†	iOS 11.2.1	Siri	√	√	24–26 [24.8]	100%	6	5
	Apple	iPhone 6s *	iOS 10.2.1	Siri	√	√	26 [26]	100%	4	12
	Apple	iPhone 6s Plus *	iOS 11.2.1	Siri	√	√	22 24–28 30–31 [25.7]	100%	7	8
	Apple	iPhone 7 Plus *	iOS 10.3.1	Siri	√	√	21 24–29 [25.3]	$\geq$ 50%	18	12
	Apple	iPhone X	iOS 11.4	Siri	√	√	20–50 [24.8]	$\geq$ 30%	56	95
	LG	Nexus 5X	Android 7.1.1	Google Now	√	√	30.7 [30.7]	100%	6	11
	Samsung	Galaxy S6 edge	Android 6.0.1	S Voice	√	√	20–38 [28.4]	$\geq$ 17%	36	56
	Samsung	Galaxy S6 edge+	Android 6.0.1	S Voice	√	√	21–22 24–40 [28.0]	$\geq$ 21%	30	35
	Huawei	Honor 7	Android 6.0	HiVoice	√	√	29–37 [29.5]	$\geq$ 17%	13	14
	Huawei	P10 Plus *	Android 7.0	HiVoice	√	√	20–25 28–30 33 [24.2]	100%	9	13
	Huawei	Mate 20 *	Android 9.0	HiAssistant	√	√	20–35 [33.8]	$\geq$ 50%	16	18
	Apple	iPad mini 4	iOS 10.2.1	Siri	√	√	22–40 [28.8]	$\geq$ 25%	91	50
	Asus	Nexus 7	Android 6.0.1	Google Now	√	√	24–39 [24.1]	$\geq$ 5%	88	87
	Apple	watch	watchOS 3.1	Siri	√	√	20–37 [22.3]	$\geq$ 5%	111	164
PC	Apple	MacBook	macOS Sierra	Siri	√	N/A	20–22 24–25 27–37 39 [22.8]	$\geq$ 76%	31	N/A
	Lenovo	ThinkPad T440p	Windows 10	Cortana	√	√	23.4–29 [23.6]	$\geq$ 35%	58	8
Speaker	Amazon	Echo	5589	Alexa	√	√	20–21 23–31 33–34 [24]	$\geq$ 20%	165	165
	JD	DingDong 2	3.1.2.169	iFlytek	√	√	28–31 [29.1]	100%	8	7
	Alibaba	Tmall Genie X1	1.4.2	AliGenie	√	√	20–25 [—]	100%	5	11
Vehicle	Audi	Q3	N/A	N/A	√	N/A	21–23 [22]	100%	10	N/A
	Tesla	Model S	8.1	N/A	×	N/A	— [—]	—	—	N/A
	Roewe	RX5	YunOS 1.1.1	Banma	√	√	— [25]	100%	10	10

‡ Prime f_c is the carrier wave frequency that exhibits highest baseband amplitude after demodulation. — No result
† Another device with the same model number.
* Experimented with the front/top microphone on the device.

the speaker. Both the device and the speaker are elevated to the same horizontal level (e.g., 10 cm above the table) to reduce mechanical coupling. All experiments except those on vehicles are conducted in our laboratory with an average background noise of 55 dB SPL (sound pressure level), and we confirm that no interfering sound exists in the 20–50 kHz frequency band. We transmit the inaudible voice commands through the powerful transmitter and observe the results on the device screen or from its vocal response.

Generally, there are multiple microphones installed on a device in order to pick up voices from multiple directions. It is a common case that all the microphones are used in speech recognition. In our experiments, we specifically test the one that shows the best demodulation effect.

Voice Commands. Two categories of voice commands are prepared for two types of attacks, activation and recognition, as listed in Tab. 2. For those systems supporting voice activation, we train them with Google TTS and try to activate them with inaudible wake word commands. To examine whether the inaudible voice commands can be correctly recognized, we prepare a set of commands to cover different types of devices. We generate each attack command with Google TTS to avoid the influence of imperfect voice synthesis.

Sound Pressure Level. Though the sound generated for attacks are inaudible to human, we nonetheless measure the sound pressure level (SPL) in decibels using a free field measurement microphone [32]. The SPL of the ultrasound is 125 dB when measured at 10 cm away from the speaker.

Attacks. In recognition attacks, the voice assistants are manually activated beforehand. While in activation attacks, physical interactions with the devices are not involved. The attacks are only considered successful and the results are only recorded when the recognized texts from the voice assistants fully match with the attacking commands.

Modulation Parameters. We argue that the modulation parameters may have an influence on the attack performance. We consider two factors in amplitude modulation: the carrier wave frequency f_c and the modulation depth. To quantify their influence, we place the devices 10 cm away from the attacking speaker and measure three parameters: (a) f_c *range*—the range of carrier wave frequencies in which recognition attacks are successful. (b) *Prime* f_c—the carrier wave frequency that exhibits the highest baseband[2] amplitude after demodulation. (c) *AM depth*—the modulation depth at the prime f_c when recognition attacks are successful.

5.3 Feasibility Results

Tab. 3 summarizes the experiment results. From Tab. 3, we conclude that DolphinAttack works with nearly all of the

[2] For simplicity, the baseband signal is a single tone at 400 Hz.

examined voice assistants and devices. In particular, the inaudible control commands can be correctly interpreted by nearly all of the tested voice assistants, and the activation commands can activate all corresponding devices. Nonetheless, the results do show that various modulation parameters are required in order to accomplish the same attacks on different voice assistants and devices. We discuss as follows.

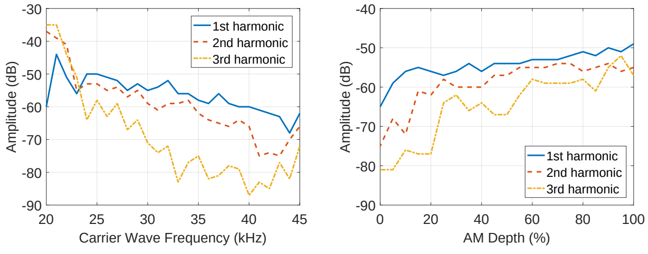
Hardware Dependence. DolphinAttack's basic principle of is to inject inaudible voice commands before digitization components. Therefore, the feasibility of DolphinAttack depends heavily on the audio hardware rather than the speech recognition systems. For example, different models of devices from the same manufacturer running the same voice assistant show great variance in the attack success rate, the maximum attack distance, and modulation parameters. This is caused by hardware variance (e.g., microphones, amplifiers, filters), which leads to variation in the digitized audio despite the same SR system.

For those devices of the same model, they exhibit similar attack parameters and results most of the cases (e.g., iPhone SE), but show slight variance as well (e.g., iPhone 6 Plus). Thus, it is feasible for an adversary to study the hardware beforehand and predict the necessary attack parameters as well as possible results on a similar device. Noticeably, we also observed in our experiments that devices with ECMs (the three vehicles) require more trials for a successful attack, which is possibly because ECMs are less sensitive to ultrasound.

SR System Influence. We find that various SR systems may handle the same audios differently. We test the voice search in Google Chrome running on an iPhone SE. The results in Tab. 3 show that the f_c range of Google Chrome overlaps with the f_c range in Siri experiment, which suggests that our attacks are indeed hardware dependent. However, the differences in f_c, AM depth, and recognition distances are resulted from the discrepancy of SR systems.

Recognition versus Activation. Various devices and SR systems can react differently to recognition and activation attacks in terms of the attack distance. For 13 devices, the activation attacks are effective at a greater distance than recognition attacks, while for the other 8 devices, the recognition attacks can be achieved further. We argue that the activation and recognition commands can show different performance when they are combined for real-life attacks, and the lowest bar determines the attack capability. We will evaluate the overall success rate of the two steps in one go in the next section.

Carrier Wave Frequency. f_c is the dominant factor that affects the attack success rate, and it also shows great variance across devices. For some devices, the f_c range within which recognition attacks are successful can be as wide as 20–50 kHz (e.g., iPhone X), or as narrow as a few single frequency points (e.g., iPhone 5s). We attribute this diversity to the difference of frequency response and frequency selectivity for these microphones as well as the nonlinearity of audio processing circuits.



(a) Carrier wave frequency.

(b) Modulation depth.

Fig. 9: Amplitude of the demodulated 400 Hz baseband signal (1st harmonic) and its higher order harmonics on Nexus 7, with varying carrier wave frequency and modulation depth.

For instance, the f_c range of Nexus 7 is from 24 to 39 kHz, which can be explained from two aspects. The f_c is no higher than 39 kHz because the frequency response of the Nexus 7 microphone is low, and a carrier higher than 39 kHz is no longer efficient enough to inject inaudible voice commands. The f_c cannot be smaller than 24 kHz because of the nonlinearity coefficients. Recall in Sec. 3.1 that frequencies other than the baseband such as $2f_m$ are also produced but not filtered, which can distort the baseband signals. We observe that the inaudible voice commands become unacceptable to SR systems when the amplitude of such frequencies are larger than the baseband. For instance, given the baseband of a 400 Hz tone, we measure the demodulated signal (i.e., the 400 Hz baseband) on a Nexus 7, and observe harmonics at 800 Hz (2nd harmonic), 1200 Hz (3rd harmonic) and even higher. As shown in Fig. 9(a), when the f_c is less than 23 kHz, the 2nd and 3rd harmonics are stronger than the 1st harmonic, which distort the baseband signal greatly and make it hard for SR systems to recognize. The *Prime* f_c that leads to the best attack performance is the frequency that exhibits both a high baseband signal and low harmonics. On Nexus 7, it is 24.1 kHz.

Modulation Depth. Modulation depth affects the amplitude of demodulated baseband signal and its harmonics, as shown in Fig. 9(b). As the modulation depth gradually increases from 0 to 100 percent, the demodulated signals become stronger, which in turn increase the SNR and the attack success rate, with a few exceptions (e.g., when the harmonics distort the baseband signal more than the cases of a lower AM depth). We report the minimum depth for successful recognition attacks on each device in Tab. 3.

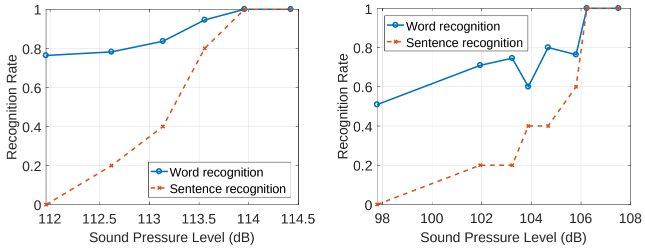
Attack Distance. The attack distance is largely determined by the power of the transmitter. With the Vifa ultrasonic speaker in Fig. 4, the maximum distance that we can achieve for both attacks is 165 cm on an Amazon Echo. We need to point out that the attack distance can be increased dramatically with more powerful transmitters. For example, later in Sec. 7 we achieve an attack distance of 19.8 m with a transmitter array while the attacks still remain inaudible. Nevertheless, the distances we report in Tab. 3 serve as an important reference for the comparison of attack feasibility on various devices. For example, the iPhone 4s that can be activated at 110 cm away is easier to attack than the iPhone 6 Plus that can only be activated at 2 cm away.

Attack Angle. Though unreported in Tab. 3, the angle between the transmitter and microphone can affect the attack performance greatly because microphones cannot pick up sounds equally from all directions. Generally, DolphinAttack works the best when attacking from a straight-up angle. We failed in attacking a Tesla vehicle be-

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TDSC.2019.2906165, IEEE Transactions on Dependable and Secure Computing

TABLE 4: The impact of background noises for sentence recognition evaluated with an Apple watch. A total of 10 trials are performed and we count a successful recognition when every word in the command is correctly recognized.

Scene	SPL (dB)	Recognition rates	
		Hey Siri	*Turn on airplane mode*
Office	55–65	10/10	10/10
Cafe	65–75	10/10	8/10
Street	75–85	9/10	3/10



(a) Samsung Galaxy S6 Edge (b) Apple watch

Fig. 10: The impact of sound pressure levels on the word and sentence recognition rates of a control command (over 10 trials), experimented on two smart devices.

cause the ECMs are placed sideways and can barely receive ultrasounds. After dismantling the microphone unit near the sunroof and exposing the microphones, we managed to inject inaudible commands.

Efforts and Challenges. We encounter a few challenges in conducting the above experiments. Apart from acquiring the devices, measuring each parameter is time-consuming and labor-intensive due to the lack of audio measurement feedback interface. For example, to measure the *Prime* f_c, we analyze the demodulation results on various devices using several audio spectrum analyzing software. For devices not supporting installing spectrum software such as Apple watch and Amazon Echo, we utilize the calling and command log playback function, and measure the audio on another relaying device.

6 IMPACT QUANTIFICATION

In this section, we evaluate the performance of `DolphinAttack` in terms of background noises, sound pressure levels, and attack distances using the powerful transmitter shown in Fig. 4. In addition, we evaluate the effectiveness of walk-by attacks with a portable transmitter and remote attacks with traditional loudspeakers.

6.1 Impact of Background Noise

Speech recognition is known to be sensitive to background noises and is recommended to be used in a quiet environment. Thus, we examine inaudible voice command injection via `DolphinAttack` in three scenarios: in an office, in a cafe, and on the street. To make the experiment repeatable, we simulate the three scenarios by playing background sounds at a chosen SPL and evaluate their impact on the recognition rates. We select an Apple watch as the attack target, and measure the background noise with a sound meter.

From Tab. 4, we conclude that background noises have a negative impact on the recognition of inaudible voice commands as well. The recognition rates of both activation command ("Hey Siri") and control command ("Turn on airplane mode") decrease with the increase of ambient noise levels. Noticeably, the activation command is recognized more times than the control command as the noise level increases. We assume this is because the activation command is shorter and has been previously learned by the SR system.

6.2 Impact of Sound Pressure Level

For both audible and inaudible sounds, a higher SPL leads to a better quality of recorded voices and thus a higher recognition rate. This is because a higher SPL always means a larger signal-to-noise ratio (SNR) for given noise levels. To explore the impact of SPLs on `DolphinAttack`, we test a control command ("Call 1234567890") on an Apple watch and a Galaxy S6 Edge smartphone. In all experiments, the speaker is positioned 10 cm away from the target device, and a microphone is placed next to the speaker to monitor the SPL.

We quantify the impact of SPLs with two granularities: *word recognition rate* and *sentence recognition rate*. Word recognition rate refers to the percentage of words that are correctly interpreted in a command. For example, if the command "Call 1234567890" is recognized as "Call 124567", the word recognition rate is 63.6 percent (7/11). Sentence recognition rate is calculated as the number of trials with 100 percent word recognition rate over 10 trials.

Fig. 10 shows the impact of the SPLs on both types of recognition rates. Not surprisingly, under the same SPL, the word recognition rates are always higher than the sentence recognition rates until both reach 100 percent. For the Apple watch, both recognition rates become 100 percent once the SPL is larger than 106.2 dB. In comparison, the minimum SPL for the Galaxy S6 Edge to achieve a 100 percent recognition rate is 113.96 dB, which is higher than that of the Apple watch. This is because the Apple watch outperforms the Galaxy S6 Edge in terms of demodulating inaudible voice commands.

6.3 Impact of Attack Distance

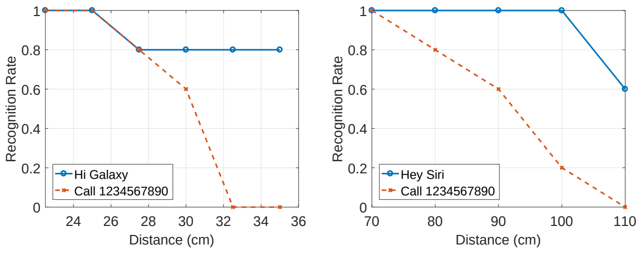
We quantify the recognition rates on two devices at various distances with two activation commands ("Hey Siri" and "Hi Galaxy") and a control command ("Call 1234567890") and show the results in Fig. 11.

In general, the sentence recognition rates of the activation command are higher than that of the control command, because the activation command contains less words than the control command. The Apple watch can be activated with a success rate of 100 percent from 100 cm away, and the Galaxy S6 Edge can be activated with 100 percent from 25 cm. We assume the difference between the two devices is because Apple watches are worn on the wrist and are designed to accept voice commands from a longer distance than a smartphone.

6.4 Evaluation of Attacks with a Portable Transmitter

In this subsection, we evaluate the effectiveness of `DolphinAttack` with a portable transmitter.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TDSC.2019.2906165, IEEE Transactions on Dependable and Secure Computing



(a) Samsung Galaxy S6 Edge (b) Apple watch

Fig. 11: The impact of attack distance on the sentence recognition rates of activation and control commands over 10 trials, experimented on two smart devices.

TABLE 5: The results of attacking an Apple watch using a Galaxy S6 Edge smartphone at 2 cm away.

f_c (kHz)	20	21	22	23	24
Word recognition rate	80%	100%	15%	100%	0%
Sentence recognition rate	8/10	10/10	0/10	10/10	0/10

Setup. We first test the portable setup using the Samsung smartphone only, and attack an Apple watch paired with an iPhone 6 Plus. The attack voice command is "Turn on airplane mode" and we perform 10 trials. We set f_c to be {20, 21, 22, 23, 24} kHz, respectively. The AM depth is 100 percent, and the sample rate is 192 kHz. The baseband signal has a limited bandwidth of 3 kHz.

Results. As shown in Tab. 5, we successfully "turned on airplane mode" on the Apple watch at the 23 kHz carrier frequency with 100 percent word and sentence recognition rates. Note that 20 kHz and 21 kHz are also successful. However, there are audible frequency leakages below 20 kHz which sound like crickets. With the increase of f_c, the Apple watch fails to recognize the voice command because the Samsung smartphone limits the sample rate on its internal speaker.

To increase the attack distance, we extend the portable setup with an off-the-shelf low-power audio amplifier module and an ultrasonic transducer, as is shown in Fig. 8. With the 3-watt amplifier module, the maximum attack distance is increased to 27 cm. Note that the attack distance can be further extended with professional transducers and more powerful amplifiers.

6.5 Evaluation of Remote Attacks with Traditional Speakers

An adversary can launch a remote attack utilizing traditional speakers in private or public areas. For example, an adversary can upload an audio or video clip to a website (e.g., YouTube) in which the inaudible voice commands are embedded. When the audio or video is decoded by a victim's computer/smartphone and played by a traditional speaker, the surrounding voice controllable devices such as Google Home, Amazon Echo, and smartphones might be controlled unconsciously. In extreme conditions, multiple devices might be attacked at the same time.

6.5.1 Feasibility Experiments with Traditional Speakers

The feasibility of remote attacks relies on whether traditional speakers can play ultrasounds embedded in regular audio or video files remotely. We argue that such requirements can be satisfied with the following observations.

TABLE 6: The sound pressure levels of four traditional speakers at five carrier wave frequencies.

Speaker	SPL (dB) at five f_c (kHz)				
	20	21	22	23	24
HiVi	84.2	86.0	85.5	89.9	88.8
JBL	75.9	71.5	61.9	51.5	30.2
Fostex	81.9	81.0	78.8	77.5	75.9
GT1188	78.8	86.0	81.0	80.0	81.0

- **Sample rate**. The sample rates of audio files and sound cards determine the maximum frequency of the signals that can be delivered to speakers, which is a half of the sample rate. The Audio Engineering Society recommends 48 kHz sample rate for most audio applications [33]. Such a sample rate allows to embed ultrasonic frequencies up to 24 kHz in an audio file, which is sufficient for attacks with low f_c (e.g., 20–21 kHz). In case of the speakers for high fidelity (HiFi) music, higher sample rates, such as 96 kHz and 192 kHz, can be used, and they support higher ultrasonic frequencies and enable a larger range of f_c for attacking various devices. In fact, many websites, smartphones, music players, and computer sound cards support audio sample rates up to 192 kHz [34], [35].
- **Frequency response of speakers**. Traditional loudspeakers are designed to produce sounds within the human hearing range (i.e., 20 Hz to 20 kHz). However, many speakers can deliver frequencies beyond 20 kHz, especially the high-quality speakers. To reproduce sounds balanced at all frequencies, a high-quality speaker is normally a combination of several types of speakers that are good at producing sounds in different frequency ranges, e.g., woofer (40–500 Hz), mid-range (250–2000 Hz), and tweeter (above 2000 Hz) speakers. Most tweeters are capable to deliver frequencies higher than 20 kHz, though not as strong as below 20 kHz.

Speakers and devices. We validate the feasibility of remote attacks with four traditional speakers: a high-end speaker from HiVi [36], a portable mini Bluetooth speaker from JBL [37], and two tweeters [38], [39] that are used in home and studio speakers. The target devices include an Apple watch, an iPhone 4s, and an Amazon Echo.

Setup. The attack commands are "Hey Siri, call 1234567890" and "Alexa, open the back door". We modulate the commands on five f_c and generate audio files in MATLAB with 96 kHz sample rate and 16-bit depth. We connect the speakers to a Samsung smartphone via the headphone jack and play the audio file. The devices are placed 10 cm away from the speakers with their microphones at the straight angle. For each speaker-device pair at each f_c, we perform 10 trials and count the number of trials that the attack commands activate the device and are correctly recognized. The sound pressure levels of the speakers used in the experiments are measured on the device side and shown in Tab. 6.

Results. As shown in Tab. 7, the traditional speakers are capable to emit inaudible voice commands and thus can be

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TDSC.2019.2906165, IEEE Transactions on Dependable and Secure Computing

TABLE 7: The experiment results on the feasibility of remote attacks. The recognition rates over a basis of 10 trials are reported with regard to attacking three devices with four traditional speakers at five carrier wave frequencies.

Speaker	**Device**	**Recognition rates at five f_c (kHz)**				
		20	**21**	**22**	**23**	**24**
HiVi	Apple watch	10/10	10/10	10/10	10/10	10/10
	iPhone 4s	10/10	10/10	10/10	0/10	0/10
	Amazon Echo	2/10	10/10	1/10	0/10	0/10
JBL	Apple watch	10/10	10/10	0/10	0/10	0/10
	iPhone 4s	0/10	9/10	0/10	0/10	0/10
	Amazon Echo	3/10	10/10	4/10	0/10	0/10
Fostex	Apple watch	10/10	10/10	10/10	10/10	10/10
	iPhone 4s	8/10	10/10	9/10	6/10	5/10
	Amazon Echo	6/10	5/10	5/10	3/10	0/10
GT1188	Apple watch	10/10	10/10	6/10	10/10	10/10
	iPhone 4s	10/10	10/10	10/10	2/10	1/10
	Amazon Echo	9/10	10/10	9/10	5/10	0/10
Average recognition rate		7.3/10	9.5/10	6.2/10	3.8/10	3.0/10

utilized for remote attacks. The highest average recognition rate is 9.5 trials over 10 trials when f_c is 21 kHz. We observe that both the speakers' capability of emitting ultrasound and the devices' capability of receiving ultrasound can affect the recognition rates. For example, the JBL speaker yields low recognition rates when f_c is above 21 kHz because its produced SPL is low, and the Apple watch is more vulnerable than the other devices by being sensitive to all tested frequencies. Nevertheless, it is feasible for an attacker to achieve a successful remote attack by choosing a proper f_c and performing repeated trials.

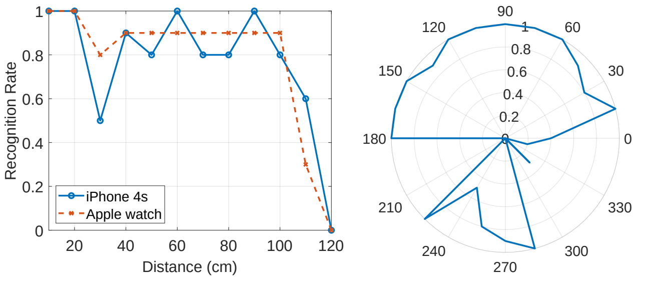
We have validated the feasibility of remote attacks in controlled conditions. Nest, we evaluate the reliability of remote attacks with the factors reflected in real-life scenarios, i.e., at variant distances, with random relative orientation between the device and speaker, and under various levels of background noises. Audio files with f_c at 21 kHz are used for the rest of evaluations.

6.5.2 Impact of Attack Distance

We attack the Apple watch and iPhone 4s with the HiVi speaker and record the activation rates with the increase of the distance between the device and the speaker. The speaker is adjusted to its maximum volume. As shown in Fig. 12(a), the recognition rates are nearly all above 8/10 for both devices when the distance is within 1 meter, and drop to 0 when the distance is above 1.2 m. The results suggest that remote attacks are likely to succeed when the speaker and device are physically close, e.g., on the same table.

6.5.3 Impact of Relative Orientation

In practice, a device can be randomly orientated with regard to speakers. To understand the impact of the orientation, we place the HiVi speaker at 24 angles around the Apple watch and set the distance to 50 cm. The only microphone on the watch is pointed at 90°. The results shown in Fig. 12(b) suggest that the attacks are successful in a wide range of angles. The recognition rates are higher than 8/10 when the speaker is in front of the microphone (0°-180°), and more than 5/10 at some angles (235°-285°) even when the speaker is behind the microphone, which is possibly caused by the reflections of ultrasound (from the wall or tabletop).



(a) Distance

(b) Relative orientation

Fig. 12: The recognition rates with regard to (a) the attack distance between the HiVi speaker and two devices, and (b) the relative orientation between the HiVi speaker and Apple watch (the only microphone on the watch is pointed at 90°).

TABLE 8: The recognition rates under six sound pressure levels of white noise.

White noise SPL (dB)	**40**	**50**	**60**	**70**	**72**	**75**
Recognition rate	10/10	10/10	10/10	10/10	1/10	0/10

6.5.4 Impact of Background Noise

A background noise can lower the recognition rates of both human voice and inaudible voice commands. We attack the Apple watch with the HiVi speaker from 30 cm away and record the recognition rates under six levels of white noise (measured on the device side) played by the JBL speaker. As shown in Tab. 8, the recognition rates are 10/10 when the noise is below 70 dB SPL, which suggests that remote attacks can be effective even in a noisy environment, e.g., a cafe.

7 Long-Range Attacks and Inaudibility

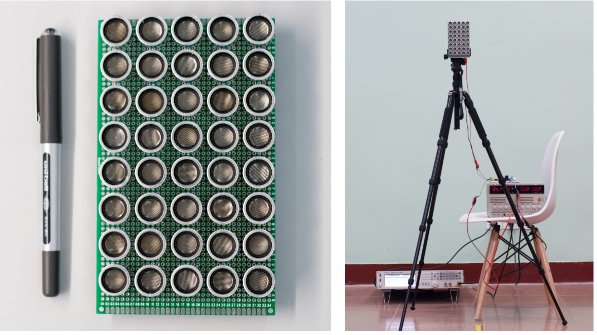
For the aforementioned attacks we focus on the feasibility and do not intentionally maximize the distance of attacks. A motivated attacker may try to launch a long-range attack locally that exceeds the distance we report in Tab. 3. In this section, we investigate the feasibility of long-range attacks by studying to what extent can a local attacker increase the attack distance and whether there are fundamental limitations on the attack performance.

7.1 Long-Range Attacks with a Transmitter Array

As the attack distance increases, the ultrasound carriers go through higher atmospheric attenuation and the recovered voice commands may not be strong enough for recognition. According to the power-law equation of acoustic attenuation, the pressure of sounds traveling a distance d is

$$P(d) = P_0 e^{-\alpha(\omega)d} \tag{3}$$

where P_0 is the pressure at the transmitter and $\alpha(\omega)$ is the attenuation coefficient dependent on the frequency ω. In order to have the same $P(d)$ as d increases, two parameters can be optimized—a lower $\alpha(\omega)$ or a higher P_0.



(a) A transmitter array (b) The long-range setup

Fig. 13: The transmitter array and experiment setup for the long-range attacks.

Transmitter Array. To achieve a higher sound pressure P_0 at the transmitter, we build a transmitter array with 40 ultrasonic transducers in parallel and drive them with a TPA3116D2-based power amplifier circuit, as shown in Fig. 13. The total cost for the transmitter array and power amplifier is \$36. We choose 25 kHz as the center frequency of the transducers for a low $\alpha(\omega)$. Compared with other common transducer frequencies such as 40 kHz, ultrasounds at 25 kHz are less attenuated and can be received by most devices as we tested in Tab. 3.

Results. Our transmitter array boosts the attack distance dramatically. At a power of 1.5 W, the voice commands modulated on ultrasounds can be correctly recognized by the Siri on an iPhone SE at 10 m away (previously 0.3 m) and on an iPhone X at 19.8 m away. Therefore, it is feasible for an attacker to launch long-range attacks outside an open window or across the street. Note that the attack commands are completely inaudible to a person either close to the transmitter array or the device being attacked. However, if we keep increasing the transmission power above 1.5 W the attack becomes audible. We look into this phenomenon and elaborate our findings in the following.

7.2 On the Boundary of Inaudibility

We observed in our experiments that increasing the transmission power beyond a threshold level could turn the attacks audible. Studying the audibility of `DolphinAttack` at high power levels is important because it determines whether the inaudible attacks can remain hidden successfully. In the following, we experimentally investigate the source of such audibility and discuss theoretical explanations.

The Source of Audibility. The inaudible commands may become audible during three stages of the signal chain: the sound source (speakers), the transmission medium (air), and the receiving system (human ears).

1) *The sound source (speakers).* The nonlinearity of amplifiers and speakers may create audible byproducts that are emitted alongside the inaudible commands [40].
2) *The transmission medium (air).* The transmission of ultrasounds in a nonlinear medium (e.g., the air) may create audible sounds during the propagation [41].
3) *The receiving system (human ears).* Ultrasonic hearing is a recognized auditory effect that allows human to perceive ultrasounds as audible sounds [42].

Since the audible perception may be caused by each one of the three stages, it is important that we quantify the amount of audible perception each stage contributes such that we pinpoint the dominant source of audibility. Since it is difficult to objectively quantify the auditory perception inside human ears and brains, we focus on quantifying the first two sources, i.e., the nonlinearity of the sound source and the transmission medium. The interesting question is which of the speakers' and the air's nonlinearity dominates the IMD and creates the most of the audible sounds. This is important to understand, because two of them will create two shapes of ranges within which the inaudible commands become audible and lead to various levels of suspicion: (1) the nonlinearity of the speakers will create an audible sphere centered at the speakers, and (2) the one from the air creates an audible range on the propagation path of the ultrasounds. Thus, the audible range created by the air's nonlinearity is less likely to cause suspicion, and only when a user is located on the line of sight between the speaker and the target, can she hear it.

Effects of Nonlinear Acoustics. When sound waves have *sufficiently large amplitudes*, their propagation in the air can no longer be modeled by the traditional linearization of fluid dynamics equations. Such a phenomenon has been well studied as a branch of physics called *Nonlinear Acoustics* [41]. In the following we give an intuitive explanation of nonlinear acoustics. A sound wave propagates through a medium as localized pressure change. The increased local pressure of the air increases its local temperature, which on the other hand also increases the local speed of sound. As a result, a sound wave travels faster during the higher pressure phase of the oscillation than during the lower pressure phase. This distorts the sound wave and affects its frequency structure. Such an effect is minimal if the sound has low amplitudes, but it turns unignorable if the amplitude is high, especially for ultrasonic waves due to their relatively high amplitude to wavelength ratio. Thus, in the long-range attacks, the ultrasounds of high amplitudes are distorted as they propagate and produce audible sounds in the air due to the effect of nonlinear acoustics. The SPLs of the audible sounds created by nonlinear acoustics can exhibit two types of change:

1) *Increase stage.* The SPLs may increase with distance first, because the distortion of ultrasounds (i.e., the generation of audible sounds) is a cumulative process with regard to the distance [43].
2) *Decrease stage.* The SPL decreases when the distance is beyond a threshold because of the dissipation of the wave energy (i.e., attenuation of ultrasounds) is dominant.

According to Berktay's solution [44] to the Westervelt Equation [41], the pressure of the self-demodulated audible sound is proportional to the square of the pressure at the transmitter, which can be simplified as

$$P_d \propto P_0^2 \tag{4}$$

Therefore, as one increases the ultrasound pressure P_0 at the transmitter for a longer attack range d, audible sounds are inevitably demodulated in the air and easily become louder, making the inaudible voice commands audible. Under the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TDSC.2019.2906165, IEEE Transactions on Dependable and Secure Computing

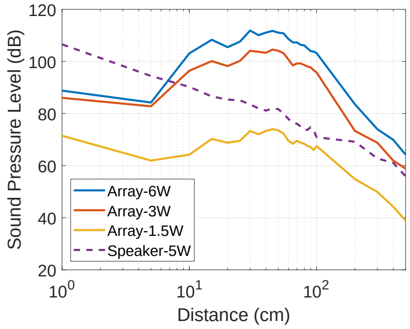


Fig. 14: The loudness of the 1 kHz sound at various distances produced by a traditional loudspeaker and the transmitter array at three power levels.

same principle, the acoustics community has built highly directional "Audio Spotlight" speakers [45], [46] that can deliver audible sounds (e.g., music and speech) to a restricted area.

Experimental Setup. To quantify the audibility from the speaker and the air, we modulated a 1 kHz tone on 25 kHz carriers and played the signals with our transmitter array at three power levels—1.5 W, 3 W and 6 W. If nonlinearity exists, a 1 kHz tone shall be heard. For comparison, we also played a 1 kHz tone from a loudspeaker at 5 W. We received the sounds with a measurement microphone [47] from 1 cm to 500 cm away and measured the sound pressure levels of signals at 1 kHz on a spectrum analyzer [48].

How to Measure Audible Sounds? Since the nonlinearity of microphones can turn ultrasounds into audible sounds, what a microphone receives can be the combination of existing audible sounds and the ones demodulated by the microphone. To overcome the challenges and to alleviate the influence of the nonlinearity of microphones, we utilized the propagation characteristics of ultrasounds, i.e., ultrasounds are highly directional. Thus, we can reduce the amount of received ultrasounds (and the amount of audible sounds demodulated by the microphone) by placing the microphone perpendicular to the sound transmission path.

Results. As a reference, we measured the SPL of the audible sounds played at 5W over a traditional speaker and observed it decreased with distance, as shown in Fig. 14. In comparison, the SPLs of the audible sounds from the ultrasonic transmitter array do not always monotonically decrease with distance. They increase first as the nonlinearity accumulates with distance and decrease as the propagation attenuation exceeds the effect of nonlinearity accumulations. From ultrasound experiments, we observe that both the speaker and the air exhibit nonlinearity and produce audible sounds at 1 kHz, and the air's nonlinearity is more dominant in producing the audible sounds, especially when the ultrasounds are at high power levels. Within 5 cm of the transmitter array, audible sounds with their SPLs higher than 70 dB are detected, and the SPLs decrease with distance, which suggests that the transmitter array emits audible sounds similar to the sounds from a traditional speaker. However, the SPLs start to increase beyond 5 cm. They reach their maxima at around 50 cm and start to decrease exponentially after 1 m. This confirms with nonlinear acoustics and means that the audible sounds are also created as the ultrasounds travel in the air. In addition, in our experiments we heard sounds mainly in a highly directional space in front of the transmitter array rather than in a spherical space around it. Such an audible perception with our ears is consistent with the microphone measurements. In either case of the source of audibility, increasing the power of ultrasounds led to louder audible sounds.

Eliminating the Audibility. To eliminate the audibility, an attacker needs to avoid the nonlinearity of the speakers and the air. Although audible sounds from the speakers is not evident in our experiments possibly due to our high-quality amplifiers and transducers, a recent work [40] has proposed a method to eliminate the audibility from speakers by emitting narrow-band ultrasounds from multiple speakers. However, regardless of whether the nonlinearity of speakers is avoided, the nonlinear acoustics in the air still exist during the propagation of ultrasounds and will produce audible commands. We leave this challenge to future work.

8 Defenses

In this section, we discuss the defense strategies to address the aforementioned attacks from both the hardware and software perspectives.

8.1 Hardware-Based Defense

We propose two hardware-based defense strategies: microphone enhancement and baseband cancellation.

Microphone Enhancement. The root cause of inaudible voice commands is that microphones can sense acoustic sounds with a frequency higher than 20 kHz while an ideal microphone should not. By default, most MEMS microphones on mobile devices nowadays allow signals above 20 kHz [49], [50]. Thus, a microphone shall be enhanced and designed to suppress any acoustic signals whose frequencies are in the ultrasound range.

Inaudible Voice Command Cancellation. Given the legacy microphones, we can add a module prior to LPF to detect `DolphinAttack` and cancel the demodulated voice commands. In particular, we can detect the signals within the ultrasound frequency range that exhibit AM modulation characteristics, and demodulate the signals to obtain the baseband. For instance, in the presence of inaudible voice command injection, besides the demodulated baseband signals $m(t)$, the recorded analog voice signals shall include the original modulated signal: $v(t) = Am(t)\cos(2\pi f_c t) + \cos(2\pi f_c t)$, where A is the gain for the input signal $m(t)$. By down-converting $v(t)$ to obtain $Am(t)$ and adjusting the amplitude, we can subtract the baseband signal. Note that such a cancellation procedure will not affect the normal operation of a microphone, since there will be no correlation between the captured audible voice signals and noises in the ultrasound range.

8.2 Software-Based Defense

Software-based defense looks into the unique features of demodulated voice commands which are distinctive from the genuine ones.

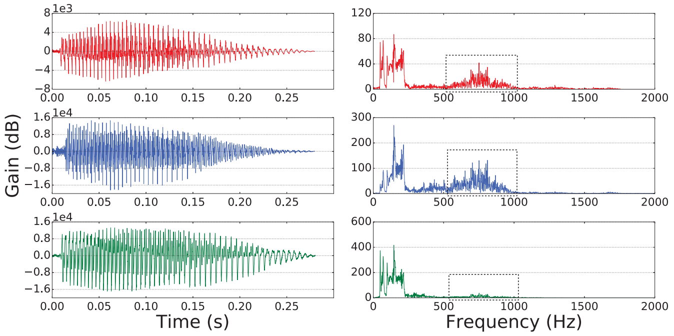


Fig. 15: Original (top), recorded (middle) and recovered (bottom) voice signals. The modulated voice command differs from both the original signal and the recorded one in the frequency range between 500 and 1000 Hz.

We analyze a voice signal generated by the Google TTS engine when it is a) in its original form, b) played and recorded, and c) modulated onto and recovered from a 25 kHz carrier. As shown in Fig. 15, the recovered signal is different from both the original signal and the recorded one in the higher frequency band ranging from 500 to 1000 Hz. Thus, we can detect `DolphinAttack` by analyzing the signal from such a frequency range. In particular, a machine learning based classifier shall detect it.

To validate the feasibility of detecting `DolphinAttack`, we utilize supported vector machine (SVM) as the classifier, and extract 15 features in the time and frequency domains from audios. We generate 12 types of a voice command ("Hey Siri"): 8 types from the NeoSpeech TTS engine and 4 types from the Selvy TTS engine. With each type, we obtain two samples: one is recorded and the other is recovered. In total, we have 24 samples. To train a SVM classifier, we use 5 recorded audios as positive samples and 5 recovered audios as negative samples. The rest 14 samples are used for testing. The classifier can distinguish the recovered audios from the recorded ones with 100 percent true positive rate (7/7) and 100 percent true negative rate (7/7). The results using a simple SVM classifier indicate that software-based defense strategies can be used to detect `DolphinAttack`.

9 Related Work

Security of Voice Controllable Systems. An increasing amount of research effort has been devoted to the security of voice controllable systems [5], [6], [11], [51], [52]. Kasmi et al. [11] introduced a voice command injection attack against modern smartphones by applying intentional electromagnetic interference on headphone cables. Mukhopadhyay et al. [51] demonstrated voice impersonation attacks on state-of-the-art automated speaker verification algorithms. Diao et al. [52] designed permission bypass attacks from a zero-permission Android application through phone speakers. Hidden voice commands [5] and Cocaine noodles [6] use audible but mangled audio commands that cannot be easily understood by human to attack speech recognition systems. `DolphinAttack` is motivated by these studies, and it is completely inaudible and imperceptible to human.

Security of Sensor-Equipped Devices. Commercial devices (e.g., smartphones, wearables and tablets) equipped with various sensors are gaining their popularity. Along with the growing trend of ubiquitous mobile devices are the security concerns. Many researchers focus on studying possible attacks against sensors on smart devices [53], [54], [55], [56], [57]. Among which, sensor spoofing (the injection of a malicious signal into a victim sensor) has attracted much attention and is considered one of the most critical threats to sensor-equipped devices [58]. Our work focuses on microphones, which is considered as one type of sensors.

Privacy Leakage Through Sensors. Michalevsky et al. [59] managed to reveal the speaker information by measuring acoustic signals with MEMS gyroscopes. Aviv et al. [60] demonstrated that accelerometers can reveal user taps and gesture-based input. Dey et al. [61] studied how to fingerprint smartphones utilizing the imperfections of on-board accelerometers, and the fingerprints can act as an identifier to track the smartphone's owner. Simon et al. [62] utilized video cameras and microphones to infer PINs entered on a number-only soft keyboard on a smartphone. Li et al. [63] can verify the capture time and location of the photos with the sun position estimated based on the shadows in the photo and sensor readings of the cameras. Sun et al. [64] presented a video-assisted keystroke inference framework to infer a tablet user's inputs from surreptitious video recordings of the tablet motion. Backes et al. [65] showed it is possible to recover what a dot matrix printer is printing based on the printer's acoustic noises. Similarly, we study how to utilize microphone vulnerabilities for security and privacy breaches.

Roy et al. [66] presented BackDoor, which constructs an inaudible acoustic communication channel between two speakers and a microphone over ultrasounds. In particular, they utilize two speakers to transmit ultrasounds at two frequencies. After passing through the microphone's non-linear diaphragm and power-amplifier, the two signals create a "shadow" in the audible frequency range, which could carry data. However, the "shadow" is a single tone instead of a voice command that consists of a rich set of tones. In comparison, we show it is possible to use one speaker to inject inaudible commands to SR systems, causing various security and privacy issues.

Our initial work on this topic appeared in [67]. This paper is an enhanced version with the following major differences: a) We greatly escalated the security threat of `DolphinAttack` by extending the attack distance from 1.7 m to 19.8 m and by proposing a more disrupting and easy-to-exploit remote attack that opportunistically utilizes a victim's traditional speakers as the transmitters. b) We thoroughly studied the audible effects that happen when transmitting ultrasounds at high power levels. c) We investigated two major factors that may affect the attack result: distortion and type of microphone, and we validated the generality of attacks on 10 more devices and 5 more types of voice assistants.

10 Conclusion

In this paper, we propose `DolphinAttack`, an inaudible attack to voice assistants and voice controllable devices. It modulates audible voice commands on ultrasonic carriers so that the command signals cannot be heard by human, but can be perceived by nonlinear hardware. With `DolphinAttack`, an adversary can attack major voice assistants including Siri, Google Now, Alexa, and etc. To avoid the abuse of `DolphinAttack` in reality, we propose two defense solutions from the aspects of both hardware and software.

References

[1] Apple, "iOS–Siri–Apple," https://www.apple.com/ios/siri/, 2017.

[2] Google, "Google Now," http://www.androidcentral.com/google-now, 2016.

[3] Amazon, "Alexa," https://developer.amazon.com/alexa, 2017.

[4] Business Insider, "Mercedes is building its own AI-powered voice assistant for the car," https://www.businessinsider.com/mercedes-building-its-own-ai-powered-voice-assistant-for-the-car-2018-1, 2018.

[5] N. Carlini, P. Mishra, T. Vaidya, Y. Zhang, M. Sherr, C. Shields, D. Wagner, and W. Zhou, "Hidden voice commands," in *Proceedings of the USENIX Security Symposium*, 2016.

[6] T. Vaidya, Y. Zhang, M. Sherr, and C. Shields, "Cocaine noodles: Exploiting the gap between human and machine speech recognition," in *Proceedings of the USENIX Workshop on Offensive Technologies*, 2015.

[7] H. Lee, T. H. Kim, J. W. Choi, and S. Choi, "Chirp signal-based aerial acoustic communication for smart devices," in *Proceedings of the IEEE International Conference on Computer Communications*, 2015.

[8] Samsung, "What is S Voice?" http://www.samsung.com/global/galaxy/what-is/s-voice/, 2017.

[9] Xdadevelopers, "HiVoice app, what is it for?" https://forum.xda-developers.com/honor-7/general/hivoice-app-t3322763, 2017.

[10] Microsoft, "What is Cortana?" https://support.microsoft.com/en-us/help/17214/windows-10-what-is, 2017.

[11] C. Kasmi and J. L. Esteves, "IEMI threats for information security: Remote command injection on modern smartphones," *IEEE Transactions on Electromagnetic Compatibility*, vol. 57, no. 6, pp. 1752–1755, 2015.

[12] C. Ittichaichareon, S. Suksri, and T. Yingthawornsuk, "Speech recognition using MFCC," in *Proceedings of the International Conference on Computer Graphics, Simulation and Modeling*, 2012.

[13] O. Viikki and K. Laurila, "Cepstral domain segmental feature vector normalization for noise robust speech recognition," *Speech Communication*, vol. 25, no. 1, pp. 133–147, 1998.

[14] STMicroelectronics, "Tutorial for MEMS microphones," www.st.com/resource/en/application_note/dm00103199.pdf, 2017.

[15] P. Horowitz and W. Hill, *The art of electronics*. Cambridge Univ. Press, 1989.

[16] J. Gago, J. Balcells, D. GonzÁlez, M. Lamich, J. Mon, and A. Santolaria, "EMI susceptibility model of signal conditioning circuits based on operational amplifiers," *IEEE Transactions on Electromagnetic Compatibility*, vol. 49, no. 4, pp. 849–859, 2007.

[17] Keysight Technologies, "N5172B EXG X-Series RF Vector Signal Generator, 9 kHz to 6 GHz," http://www.keysight.com/en/pdx-x201910-pn-N5172B, 2017.

[18] Avisoft Bioacoustics, "Ultrasonic dynamic speaker vifa," http://www.avisoft.com/usg/vifa.htm, 2017.

[19] Analog Devices, "ADMP401: Omnidirectional microphone with bottom port and analog output," http://www.analog.com/media/en/technical-documentation/obsolete-data-sheets/ADMP401.pdf, 2011.

[20] CMU Speech Group, "Statistical parametirc sythesis and voice conversion techniques," http://festvox.org/11752/slides/lecture11a.pdf, 2012.

[21] Google, "Cloud text-to-speech," https://cloud.google.com/text-to-speech/, 2018.

[22] Selvy Speech, "Demo-Selvy TTS," http://speech.selvasai.com/en/text-to-speech-demonstration.php, 2017.

[23] Baidu, "Baidu translate," http://fanyi.baidu.com/, 2017.

[24] Sestek, "Sestek TTS," http://www.sestek.com/, 2017.

[25] NeoSpeech, "Text-to-speech," http://www.neospeech.com/, 2017.

[26] Innoetics Text-to-Speech Technologies, "Innoetics text-to-speech," https://www.innoetics.com/, 2017.

[27] Vocalware, "Vocalware TTS," https://www.vocalware.com/, 2017.

[28] CereProc, "Cereproc text-to-speech," https://www.cereproc.com/, 2017.

[29] Acapela Group, "Acapela text to speech demo," http://www.acapela-group.com/, 2017.

[30] From Text to Speech, "Free online TTS service," http://www.fromtexttospeech.com/, 2017.

[31] Jinci Technologies, "Open structure product review," http://www.jinci.cn/en/goods/112.html, 2017.

[32] Cry Sound, "CRY343 free field measurment microphone," http://www.crysound.com/product_info.php?4/35/63, 2017.

[33] A. E. Society, "Aes5-2008 (r2013): Aes recommended practice for professional digital audio - preferred sampling frequencies for applications employing pulse-code modulation," http://www.aes.org/publications/standards/search.cfm?docID=14, 2013.

[34] S. Kieldsen, "Why you should be pumped about (and just a bit sceptical of) hi-res audio," https://www.stuff.tv/features/why-you-should-be-pumped-about-and-just-bit-sceptical-hi-res-audio, 2014.

[35] Sony, "2016 high-res audio trends," https://www.sony.com/electronics/2016-hi-res-audio-trends, 2016.

[36] HiVi Inc., "swans m50w," http://www.swanspeaker.com/product/htm/view.asp?id=443, 2018.

[37] Harman International Industries, "Jbl go," https://www.jbl.com/JBL+GO.html, 2018.

[38] madisound, "Fostex ft17h horn super tweeter," https://www.madisoundspeakerstore.com/bullet-tweeters/fostex-ft17h-horn-super-tweeter/, 2018.

[39] Goldwood Sound Inc., "Gt-1188 piezo horn tweeter or midrange driver," http://www.goldwoodparts.com/gt-1188.shtml, 2018.

[40] N. Roy, S. Shen, H. Hassanieh, and R. R. Choudhury, "Inaudible voice commands: The long-range attack and defense," in *Proceeding of the 15th USENIX Symposium on Networked Systems Design and Implementation*. USENIX Association, 2018, pp. 547–560.

[41] M. F. Hamilton, D. T. Blackstock *et al.*, *Nonlinear acoustics*. Academic press San Diego, 1998, vol. 1.

[42] T. Nishimura, S. Nakagawa, T. Sakaguchi, and H. Hosoi, "Ultrasonic masker clarifies ultrasonic perception in man," *Hearing research*, vol. 175, no. 1-2, pp. 171–177, 2003.

[43] L. Bjørnø, "Introduction to nonlinear acoustics," *Physics Procedia*, vol. 3, no. 1, pp. 5–16, 2010.

[44] H. Berktay, "Possible exploitation of non-linear acoustics in underwater transmitting applications," *Journal of Sound and Vibration*, vol. 2, no. 4, pp. 435–461, 1965.

[45] F. J. Pompei, "The use of airborne ultrasonics for generating audible sound beams," in *Proceedings of the Audio Engineering Society Convention 105*. Audio Engineering Society, 1998.

[46] Holosonics, "Audio spotlight by holosonics," https://www.holosonics.com/, 2018.

[47] GRAS, "Gras 46be 1/4" ccp free-field standard microphone set," https://www.gras.dk/products/measurement-microphone-sets/constant-current-power-ccp/product/143-46be, 2018.

[48] Keysight, "N9010b exa signal analyzer," https://www.keysight.com/en/pdx-2641683-pn-N9010B/, 2018.

[49] STMicroelectronics, "MP23AB02BTR MEMS audio sensor, high-performance analog bottom-port microphone," http://www.mouser.com/ds/2/389/mp23ab02b-955093.pdf, 2014.

[50] Knowles, "SPU0410LR5H-QB Zero-Height SiSonicTM Microphone," http://www.mouser.com/ds/2/218/-532675.pdf, 2013.

[51] D. Mukhopadhyay, M. Shirvanian, and N. Saxena, "All your voices are belong to us: Stealing voices to fool humans and machines," in *Proceedings of the European Symposium on Research in Computer Security*, 2015.

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

[53] D. Foo Kune, J. Backes, S. S. Clark, D. Kramer, M. Reynolds, K. Fu, Y. Kim, and W. Xu, "Ghost talk: Mitigating EMI signal injection attacks against analog sensors," in *Proceedings of the IEEE Symposium on Security and Privacy*, 2013.

[54] C. Yan, W. Xu, and J. Liu, "Can you trust autonomous vehicles: Contactless attacks against sensors of self-driving vehicle," *DEF CON*, 2016.

[55] R. M. Ishtiaq Roufa, H. Mustafaa, S. O. Travis Taylora, W. Xua, M. Gruteserb, W. Trappeb, and I. Seskarb, "Security and privacy vulnerabilities of in-car wireless networks: A tire pressure monitoring system case study," in *Proceedings of the USENIX Security Symposium*, 2010.

[56] Y. Son, H. Shin, D. Kim, Y.-S. Park, J. Noh, K. Choi, J. Choi, and Y. Kim, "Rocking drones with intentional sound noise on gyroscopic sensors," in *Proceedings of the USENIX Security Symposium*, 2015.

[57] Y. Shoukry, P. Martin, P. Tabuada, and M. Srivastava, "Non-invasive spoofing attacks for anti-lock braking systems," in *Proceedings of the International Workshop on Cryptographic Hardware and Embedded Systems*, 2013.

[58] H. Shin, Y. Son, Y. Park, Y. Kwon, and Y. Kim, "Sampling race: Bypassing timing-based analog active sensor spoofing detection on analog-digital systems," in *Proceedings of the USENIX Workshop on Offensive Technologies*, 2016.

[59] Y. Michalevsky, D. Boneh, and G. Nakibly, "Gyrophone: Recognizing speech from gyroscope signals," in *Proceedings of the USENIX Security Symposium*, 2014.

[60] A. J. Aviv, B. Sapp, M. Blaze, and J. M. Smith, "Practicality of accelerometer side channels on smartphones," in *Proceedings of the Computer Security Applications Conference*, 2012.

[61] S. Dey, N. Roy, W. Xu, R. R. Choudhury, and S. Nelakuditi, "Accelprint: Imperfections of accelerometers make smartphones trackable." in *Proceedings of the Network and Distributed System Security Symposium*, 2014.

[62] L. Simon and R. Anderson, "PIN skimmer: Inferring PINs through the camera and microphone," in *Proceedings of the ACM Workshop on Security and Privacy in Smartphones & Mobile Devices*, 2013.

[63] X. Li, W. Xu, S. Wang, and X. Qu, "Are you lying: Validating the time-location of outdoor images," in *Proceedings of the International Conference on Applied Cryptography and Network Security*, 2017.

[64] J. Sun, X. Jin, Y. Chen, J. Zhang, Y. Zhang, and R. Zhang, "Visible: Video-assisted keystroke inference from tablet backside motion," in *Proceedings of the Network and Distributed System Security Symposium*, 2016.

[65] M. Backes, M. Dürmuth, S. Gerling, M. Pinkal, and C. Sporleder, "Acoustic side-channel attacks on printers," in *Proceedings of the USENIX Security Symposium*, 2010.

[66] N. Roy, H. Hassanieh, and R. Roy Choudhury, "Backdoor: Making microphones hear inaudible sounds," in *Proceedings of the International Conference on Mobile Systems, Applications, and Services*, 2017.

[67] G. Zhang, C. Yan, X. Ji, T. Zhang, T. Zhang, and W. Xu, "Dolphinattack: Inaudible voice commands," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2017, pp. 103–117.