When Evil Calls: Targeted Adversarial Voice over IP Network

Han Liu
Washington University in St. Louis
St. Louis, USA
h.liu1@wustl.edu

Zhiyuan Yu
Washington University in St. Louis
St. Louis, USA
yu.zhiyuan@wustl.edu

Mingming Zha
Indiana University Bloomington
Bloomington, USA
mzha@iu.edu

XiaoFeng Wang
Indiana University Bloomington
Bloomington, USA
xw7@indiana.edu

William Yeoh
Washington University in St. Louis
St. Louis, USA
wyeoh@wustl.edu

Yevgeniy Vorobeychik
Washington University in St. Louis
St. Louis, USA
yvorobeychik@wustl.edu

Ning Zhang
Washington University in St. Louis
St. Louis, USA
zhang.ning@wustl.edu

ABSTRACT

As the COVID-19 pandemic fundamentally reshaped the remote life and working styles, Voice over IP (VoIP) telephony and video conferencing have become a primary method of connecting communities together. However, little has been done to understand the feasibility and limitations of delivering adversarial voice samples via such communication channels.

In this paper, we propose TAINT - Targeted Adversarial Voice over IP Network, the first targeted, query-efficient, hard label blackbox, adversarial attack on commercial speech recognition platforms over VoIP. The unique channel characteristics of VoIP pose significant new challenges, such as signal degradation, random channel noise, frequency selectivity, etc. To address these challenges, we systematically analyze the structure and channel characteristics of VoIP through reverse engineering. A noise-resilient efficient gradient estimation method is then developed to ensure a steady and fast convergence of the adversarial sample generation process.

We demonstrate our attack in both over-the-air and over-the-line settings on four commercial automatic speech recognition (ASR) systems over the five most popular VoIP Conferencing Software (VCS). We show that TAINT can achieve performance that is comparable to the existing methods even with the addition of VoIP channel. Even in the most challenging scenario where there is an active speaker in Zoom, TAINT can still succeed within 10 attempts while staying out of the speaker focus of the video conference[1].

CCS CONCEPTS

• **Security and privacy → Software and application security**; • **Computing methodologies → Machine learning**.

KEYWORDS

speech recognition; audio adversarial attacks; VoIP network

ACM Reference Format:
Han Liu, Zhiyuan Yu, Mingming Zha, XiaoFeng Wang, William Yeoh, Yevgeniy Vorobeychik, and Ning Zhang. 2022. When Evil Calls: Targeted Adversarial Voice over IP Network. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security (CCS '22), November 7–11, 2022, Los Angeles, CA, USA.* ACM, New York, NY, USA, 15 pages. https://doi.org/10.1145/3548606.3560671

1 INTRODUCTION

Intelligent voice control (IVC) devices are playing an increasingly important role as a convenient human-computer interface in our day-to-day life. Driven by rapidly developing speech recognition techniques, they can effectively interpret and execute voice commands such as unlocking the doors and making online payments [4, 7]. In 2021, over 132 million people in the United States use voice assistant in their daily lives, and the number of voice assistant users is predicted to reach 135.6 million in 2022 [19]. The global market of voice assistant was valued at USD 5.0 billion in 2020 and is projected to reach USD 50.9 billion by 2028, at a compound annual growth rate of 30% from 2021 to 2028 [53]. On the other hand, there is a growing concern on the security of these systems, since deep neural networks (DNN), the driving technology for state-of-the-art voice assistants, has been shown to be vulnerable to adversarial inputs. In particular, prior work has demonstrated that attackers can inject malicious commands using an adversarial piece of audio that is imperceptible or unsuspicious to humans but recognizable to DNN [32, 66, 69]. However, although varying in the assumption of the attacker's knowledge [27, 36, 54], most of the existing work primarily focused on transmitting adversarial audio over-the-line and over-the-air, and delivering attacks over VoIP networks remains less explored [29]. An exception is the work by Abdullah et al. [28] exploring the feasibility of untargeted adversarial audio over telephony to prevent AI-based mass surveillance.

Changing Threat Landscape Due to COVID-19 Pandemic: The COVID-19 pandemic has forever changed us. Modern communication technologies, such as Voice-over-IP (VoIP) telephone

[1] Attack demos against Google Assistant, Amazon Echo, Microsoft Cortana, as well as adversarial audio samples of different lengths and source code of the project are available on the website: https://sites.google.com/view/targeted-adversarial-voip.

CCS '22, November 7–11, 2022, Los Angeles, CA, USA
© 2022 Copyright held by the owner/author(s).
ACM ISBN 978-1-4503-9450-5/22/11.
https://doi.org/10.1145/3548606.3560671

and video conferencing, have become a major mechanism to connect communities [51]. It is estimated that there are 204.8 billion VoIP users worldwide in 2020 [24]. Since the pandemic, Zoom has reported 350 million daily meeting participants, Microsoft Teams reported 75 million unique daily active users, and Google Meet adds roughly 2 million new users each day and hit over 100 million daily meeting participants [2, 11, 17]. While in-person activity remains an irreplaceable component in our daily lives, the rapid emergence of virtualized communities due to the pandemic is here to stay [20], with many companies offering remote working options [6]. An attacker can now misuse this new form of communication to deliver adversarial audio to an intelligent voice assistant miles away. It is, therefore, crucial to understand the feasibility as well as limitations of adversarial voice attacks over these communication channels.

TAINT: Targeted Adversarial Voice over IP Net-work: We present TAINT, the first targeted, query-efficient, hard label black-box, adversarial attack on commercial speech recognition platforms over VoIP networks. Using TAINT, an adversary can generate adversarial audio that can be injected into the VoIP channel to mislead the transcription service over-the-line or cause unintended voice commands on the participant's ASR over-the-air via the victim's computer speaker. To explore the feasibility of the proposed attack, we use TAINT to attack four commercial ASR systems (Google Assistant, Microsoft Cortana, Amazon Echo, IBM ASR) over the five most popular VoIP conferencing software applications: Zoom, Microsoft Teams, Skype, Webex, and Google Meet. TAINT can achieve a close to 100% success rate with an average signal-to-noise (SNR) level of 16.69dB against Google Speech-to-text API in fewer than 1500 queries over VoIP among five VCSs. In a more challenging setting involving interference from an active speaker, our attack can still succeed within 10 attempts without being highlighted as the active speaker by the VCS. While targeted adversarial voice is well studied in the over-the-line and over-the-air settings [33, 35, 54, 66], noise and distortion of the audio by the VoIP channel pose significant new challenges. As shown in our preliminary work in Section 3, the adversarial audio search process fails to converge to the originally intended SNR level when directly applying the existing techniques. Our attack scenario also calls for the additional consideration for perceptibility of the attack due to the popular *speaker highlight* feature in most VoIP software solutions. It is also important that the generated samples can trigger ASR in common household environments. In summary, we make the following contributions:

1) Reverse Engineering of Voice-over-IP Software Solutions and Characterization of the Acoustic Channels: To understand the root cause behind the significant degradation in the performance of the adversarial samples, we reverse-engineered the top ten most widely used VCS solutions to understand the impact of the channel on voice. We identified three common voice processing steps that can significantly impact both the optimization process to generate adversarial samples, and the robustness of the generated attacks: a) signal degradation caused by lossy compression of codec, b) frequency selectivity caused by noise suppressors and high pass filters, and c) random channel noise caused by packet loss, jitter, etc. This informed our design of the adversarial VoIP sample.

2) Tackling the Impacts from VoIP in Adversarial Audio Generation: Observing that existing genetic-algorithm-based techniques can take significantly longer to converge under the VoIP channel, we take a greedy approach and make use of momentum gradient descent. To tackle the lack of gradient information in the hard-label black-box setting, we borrow the concept of random vector probing from the image domain, and couple it with an adaptive learning rate to further accelerate convergence. However, since channel noise can nullify the theoretical gain of momentum, we leverage recursive momentum. Lastly, leveraging the frequency selection insight from our channel analysis, TAINT further guides perturbation generation to avoid the frequency filters in the channel.

3) Tackling Imperceptibility and Cyber/Physical Interference: An unavoidable by-product of aggressive greedy search is a potentially sub-optimal solution. This problem manifests in TAINT in the form of generating a less imperceptible adversarial sample. To mitigate this, we apply the principle of psychoacoustic hiding in source audio selection and target audio injection process. To avoid attack attribution, our channel analysis additionally explores the characteristics of the speaker highlighting system, finding that audio volume and audio length are the key factors. Our mitigation strategy uses this insight to enable the injection of audio hiding in the presence of an active speaker. To further mitigate the interference from the physical world (e.g., folks talking), we embed multiple attempts in organic environment noise that is common to video conferencing or train the sample to anticipate such interference.

4) Tackling Poor Network Connectivity: Based on our channel analysis, the audio channel can deteriorate significantly when the network connection is poor. In this case, the adversarial sample trained for the normal network connectivity may no longer work due to the adaptive filtering mechanisms in VCS. To tackle this challenge, we design an alternative model to capture and estimate the channel effect on the transmitted audio, which enables us to generate adversarial samples adapted to various poor network conditions without having to establish an actual network connection.

2 SYSTEM OVERVIEW AND THREAT MODEL

System Overview: The system model is shown in Figure 1. Different from the existing works [35, 66], the attacker delivers the adversarial audio via the VoIP/video conferencing, such as Zoom [26], Microsoft Teams [16], and Skype [21]. Since the VoIP channel introduces an additional layer of audio processing, the attacker has to pick the appropriate adversarial example (AE) to balance stealthiness (probability of attribution) and attack success rate, based on his/her estimation of the channel characteristics. AEs for each level of VoIP-based voice degradation are trained ahead of time, and the details on the channel characteristics are discussed in Section 4. To launch an attack, the adversary has to train an AE ahead of time, assuming certain levels of network connectivity. During a video conference, the attacker plays the AE in the video conference to launch the attack. The target of the AE attack can be either an over-the-line ASR system as in [28] or an over-the-air ASR system, such as Amazon Echo, as in [36, 69]. In over-the-line attacks, AE injected by Mallory is directly delivered to back end transcription service. In over-the-air attack, AE injected into the VoIP conference is played by the victim's computer speaker, then propagated to the target ASR over-the-air. Therefore, the attack in this work cannot

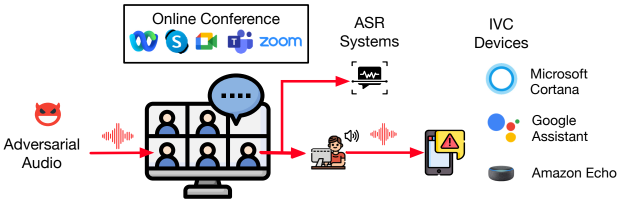


Figure 1: System model.

Table 1: Comparison of related adversarial audio attacks.

Attacks	Gradient	Conf[†]	Targeted	VoIP/Tele[‡]	Zoom SR/SNR[★]	Query[∓]
CmdSong[66]	✓		✓		0/10 / NA	1000
Metamorph[35]	✓		✓		0/10 / NA	1000
Yakura et.al.[62]	✓		✓		0/10 / NA	1000
Hidden Voice[27]			✓		1/10 / NA	5000
Abdullah et.al.[28]				✓	0/10 / NA	5000
Taori et.al.[60]		✓	✓		0/10 / NA	300000
Devil’s Whisper[36]		✓	✓		0/10 / NA	1500
OCCAM[69]			✓		10/10 / 9.42	30000
TAINT			✓	✓	**10/10 / 16.01**	**1500**

[†]: "Conf" means confidence scores. [‡]: "VoIP/Tele" means attacks are delivered over VoIP or Telephony. [★]: "Zoom SR / SNR" means the success rate and SNR of the attack when applying the Zoom channel to existing techniques. It is worth noticing that OCCAM [69] and Abdullah [27] start with correct transcription to optimize. Also, SNR in [27] cannot be calculated since it is the signal processing attack. [∓]: Query means the total number of queries we test with the Zoom channel. We use the number of queries advertise in the original paper. We modified [27] for VCS. For [28], we use the same queries as [27].

be delivered to ASR if the computer speaker is not in the same physical space as the target ASR or is not used at all. This long chain of transmission also poses new challenges in stealthiness and attribution, which are discussed later in this section.

Attack Goal: The goal of the attacker is to cause misinterpretation of audio to a targeted phrase in the ASR while surviving the VoIP channel noise and distortions. The attack can be used to mislead over-the-line speech-to-text API to prevent AI-based mass surveillance or VoIP conference transcription. It can also be used to attack ASRs in the VoIP participant’s working environment by injecting the AE into the conference and then via over-the-air from the victim computer speaker to the ASR, causing attacker-controlled smart home actions, such as opening the garage door or activating unintended programs, such as voice-recording Alexa skills [3].

Attacker Knowledge on Target ASR – Blackbox: We assume a black-box setting, where the parameters or the architecture of the target speech recognition models are unknown to the attacker, and one can only get the final transcription. We also assume the corresponding online Speech-to-Text API services, e.g., Google Cloud Speech-to-Text, are open to the public. We further make the same assumption as [36] that, for the same platform, the ASR system used to provide online speech API service is similar to the one used for the voice assistant devices.

Adversarial Sample Generation and VoIP Channel Quality Estimation: We assume that the attacker can estimate the VoIP channel conditions to determine the appropriate balance between stealthiness and success rate of the AE. For example, in Zoom, the network condition can be estimated using the network/connection diagnostics tools at the endpoints. If this information is not available, it is often possible to estimate the communication channel data rate by observing the audio quality in the meeting. Though such estimation of the channel conditions is not necessary for the attacker, the more accurate this estimate is, the better we can fine-tune the adversarial audio to be imperceptible.

Attack Attribution in Video Conferencing: An important concern in designing attacks via VoIP video conferencing is attack attribution, that is, the ability of other VCS users to identify the attacker. However, we have found that avoiding attribution is not as difficult as expected. Using Zoom as an example, many meetings do not require user authentication to join: only the link with a password embedded in the URI is needed. Thus, a malicious user can simply join the meeting under an alias. Furthermore, contrary to an expectation that playing audio will force the window to focus on the speaker, we show that it is possible to play music, adversarial car horn, baby cry, etc., at low volume without being focused on the VCS, especially when another user is speaking in the meeting.

3 EXISTING WORK AND MOTIVATION

Existing Literature: In general, the broad threat landscape of ASR systems can be categorized into three types of attacks: miscellaneous attacks, signal processing attacks, and optimization attacks [29]. Miscellaneous attacks often exploit hardware imperfection to inject signals in an in-band or out-of-band [63, 64, 68] manner. On the other hand, signal processing and optimization attacks focus on the software component, where they perturb original audio by targeting the pre-processing stage [27, 28, 32] and the DNN-based recognition phase [35, 36, 54, 60, 62, 66, 69], respectively. Our work falls into the category of optimization attacks, where the attacker leverages gradient or decision information from DNN models to generate perturbations added to the original audio.

In the past few years, existing optimization attacks have mainly focused on delivering crafted adversarial examples either over-the-air or over-the-line. Earlier works are mostly over-the-line attacks, where AEs are directly fed into APIs [33, 60]. However, the attack scenarios for such attacks are limited in practice. Thus over-the-air attacks are brought up to improve real-world feasibility [35, 62, 66]. With a leap from cyber to the physical domain, these over-the-air attacks generally play adversarial audio via speakers, which then travel through the air to reach the target ASR. Recently, there are also research interests to deliver AE over telephony networks [28].

Related Work Comparison: A detailed comparison of state-of-the-art attacks to our work is summarized in Table 1. Aside from OCCAM [69], only TAINT tackle the more restrictive hard label black-box setting, while other works require gradient and confidence information. Furthermore, only [28] and our work tackle the impacts from the VoIP/Telephony network, the channel noise poses significant new challenges compared to traditional over-the-air/line setting. Therefore, our work is closely related to two previous works [28] [69]. In [28], they manipulate audio at the level of phonemes, in an effort to disrupt machine transcription, but sound unchanged to humans. While their adversarial examples can survive telephony networks, their attack is designed as *untargeted.* In this work, we aim to advance research in this direction, by developing techniques to generate *targeted* samples. Furthermore, due to the unique setting in VoIP conferencing, TAINT also needs to take attribution avoidance (staying out of speaker focus) into consideration. Lastly, we also explored over-the-air attack after the VoIP channel, which was not explored in [28]. Another closely related work is [69]. While both our work and [69] consider hard label black-box attacks, a key differentiation of our work is the need to consider the VoIP transmission channel, which presents non-trivial challenges to adversarial sample generation due to noise and distortion. As discussed later in this section, using the same number of iterations, the SNR (a measurement for imperceptibility) of the output sample decreased by 39%.

Preliminary Experiments and Motivation: Quantitatively, we begin the exploration with the direct application of existing techniques to tackle the noise and distortion from the VoIP channel. Two laptops were set up in the lab to create a Zoom meeting, we inserted a step of passing the audio via Zoom through audio sharing and audio recording all digitally, before sending the sample to the speech-to-text API. The results are shown in Table 1. For each technique, we execute the advertised iterations in the paper and use that sample for testing. For most of the existing methods, none of the 10 generated samples succeed in the attack. The only exception is OCCAM [69], which starts with a working sample and then adds noise to improve imperceptibility. While the adversary audio did succeed, the sample SNR is at 9.42 with VoIP channel as compared to the 15.33 in [69] without VoIP channel. We've also implemented the hidden voice commands, which doesn't work well in VCS. To summarize, existing approaches either do not converge to a specific SNR level or generate less effective adversarial samples. The noise and distortion of the audio signal by the VoIP channel demand a deeper understanding of the characteristics of VoIP communication to inspire new designs for adversarial audio generation.

4 VOIP CHANNEL SYSTEMIZATION

VoIP is one of the most prominent and fastest-growing telecommunication services based on an Internet protocol suite [55]. We manually reverse engineering the various popular VCS software to understand the common elements in the software designs. The transmitter side often consists of hardware and software-based codec. The receiver side often consists of three modules: hardware interface, pre-processing module, and sound effects module. As a result, when the signal is received, processed, encoded, transmitted, and decoded, many added perturbations for adversarial manipulation can be significantly impacted by these processing modules. As a result, the impact on the audio quality from the VoIP channel is often non-trivial, especially when the network connection is unstable. To have a better understanding of the channel to guide our design, we conducted a measurement case study on the five most popular commercial video conferencing software [1]: Zoom, Microsoft Teams, Cisco Webex, Google Meet, and Skype. Our approach toward channel characterization examines the impact of network connection on three aspects, audio quality, audio frequency selectivity, and channel noise.

Audio Quality: The connection between audio quality and network performance is important: poor network performance can be problematic for real-time VoIP conversations [46], which reduces the effectiveness of adversarial samples when delivered through such channel.

Collection of Data: Three key network metrics are selected because of their impacts on VoIP audio quality: bandwidth, latency, and packet loss [46]. Network shaping tools, *tc* [22] and *Network Link Conditioner* [18], were used to simulate various network conditions from the downlink aspect, since that's the portion of the delivery channel not controllable by the attacker. We follow the setting of [48] to conduct our measurement, ten samples were collected for each network condition. To measure the voice quality, Virtual Speech Quality Objective Listener (ViSQOL) was used to compute Mean Opinion Score - Listening Quality Objective (MOS-LQO). This metric was developed to measure a spectro-temporal similarity between a reference and a test speech signal [46]. The score ranges from 1 (worst) to 5 (best).

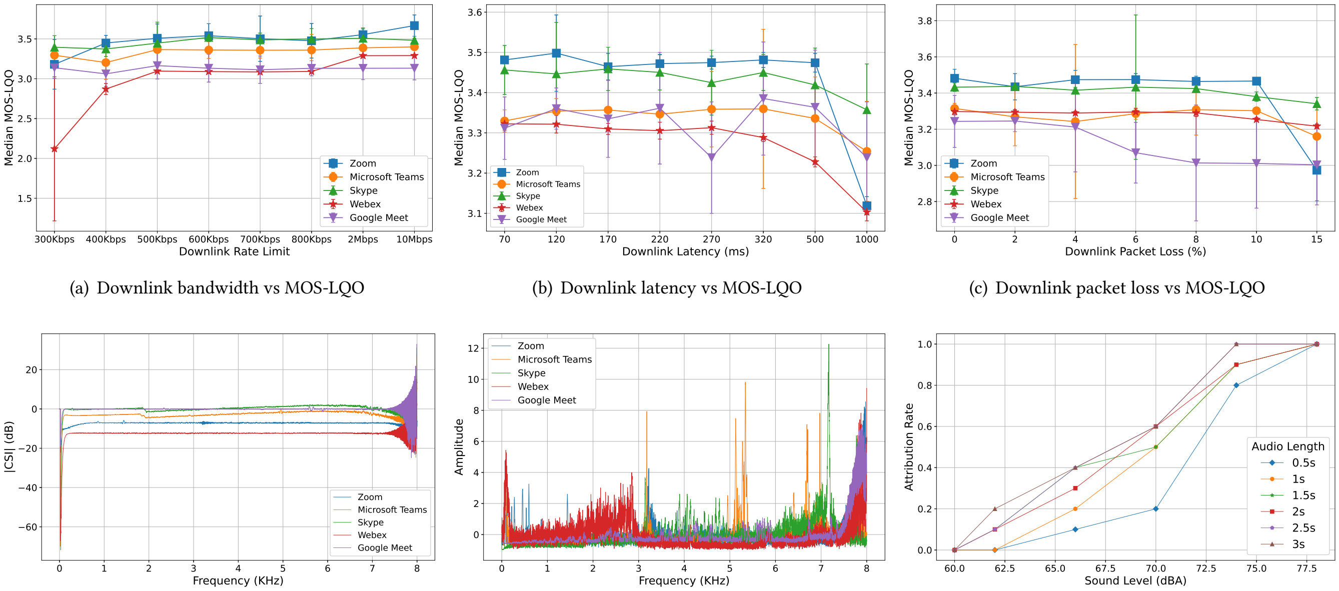
Measurement and Interpretation: The audio quality under different downlink bandwidth is given in Figure 2(a). The error bar represents the 95% confidence interval. When the bandwidth is limited (e.g. below 500Kbps), there is a significant decrease in the audio quality. The increase in variance is due to repeated reestablishment of the connections. As shown in Figure 2(b), the overall trend of audio quality remains relatively stable then drops significantly when the latency is over 220ms. As shown in Figure 2(c), the audio quality remains unaffected under a relatively small packet loss (e.g., 4%), this may be due to the VCS will allocate additional bandwidth for forward error correction (FEC) [48]. However, when the packet loss increases, the audio quality in most VCSs is also observed to noticeably decrease.

Insights: There are two key observations from these experiments. First, when bandwidth condition is above a certain threshold, audio quality is very consistent. This is consistent with our experience with VoIP video conferencing. In this case, it is possible to pretrain adversarial sample due to the stability of the channel. Second, when network connection is poor, audio quality degrades sharply. It becomes much more difficult to prepare adversarial samples ahead of time, thus motivating our design of the channel surrogate model.

Frequency Selectivity: Reverse engineering shows existing VoIP filters out certain frequencies for performance, but this may impact the effectiveness of the adversarial sample.

Collection of Data: Following [35], channel state information (CSI) is used to visualize the frequency response of the channel, which is defined as $FFT(y(t))/FFT(x(t))$. A swept sine wave [41] is sent via audio sharing feature to avoid over the air signal distortion. Frequency ranges from 20Hz to 20KHz, and the audio is recorded and then downsampled into 8 KHz segments to analyze the channel effect (e.g., DeepSpeech2 uses this range).

Measurement and Interpretation: As shown in Figure 2(d), different platforms share similar frequency selectivity properties: suppressed in the low-frequency region and the high-frequency range, relatively flat in the middle. However, for the mid-frequency range, the curve is not completely flat. The CSI of Skype has a sudden drop in the frequency range between 1.8KHz to 2KHz, Microsoft Teams are also observed with a similar phenomenon. The suppression in the low-frequency range may be due to the high pass filter in the sound effects module. The sudden drop in the middle frequency range is due to the noise suppression, which may recognize the



(a) Downlink bandwidth vs MOS-LQO

(b) Downlink latency vs MOS-LQO

(c) Downlink packet loss vs MOS-LQO

(d) VoIP channel CSI

(e) Normalized std.dev of VoIP audio amplitude

(f) Speaker focus vs sound level and audio length

Figure 2: Results of channel analysis with regarding to audio quality, frequency selectivity, channel noise, and speaker highlight.

high-frequency tone as noises with a high probability, then the sound level is suppressed. The suppression in the high-frequency range may be due to the low pass filtering deployed for preventing the aliasing in the down-sampling process [59].

Insight: Due to the frequency selectivity of the channel, the perturbation has to be adjusted accordingly to different frequency ranges, this could have a significant impact on the adversarial sample on both the success rate of the attack and the imperceptibility of the attack. As a result, it is important to take frequency selectivity into consideration in sample generation.

Random Channel Noise: It is important to understand the channel noise because adversarial perturbation can be significantly impacted, even nullify by any random noise. As a matter of fact, there is work demonstrating that a small amount of additive Gaussian noises can effectively defend against AEs [49].

Data Collection: Ten audio clips are recorded on each platform under the same network bandwidth, 100Mbps. The standard deviation of amplitude *FFT* at each frequency point is normalized by subtracting each of the data with its mean, then dividing by its standard deviation. This way, data from different platforms is scaled into the same range without affecting the original data distribution.

Measurement and Interpretation: Shown in Figure 2(e), the dynamic fluctuations of *FFT* indicate random noise added to the audio during the transmission via the channel. It shows that noises are not evenly distributed over frequency bands. However, in most regions, they fluctuate around zero mean except in the region of impulse noises. This is counter-intuitive initially, since the adversarial audio is digitally injected into the VoIP. However, upon deeper inspection, we found that many components in the system can introduce random noises. For example, the analog signal in the microphone can introduce reactive response in the audio processing functions such as noise reduction and echo cancellation. Even if we completely take them out of the picture, the long-term and short-term prediction of the adaptive rate codec deployed in the Video Conferencing Software (VCS) is sensitive to small variations among the network packets, and will produce variations in the decompressed audios.

Insight: It could take thousands of queries to generate a single adversarial sample using query-based methods. This is expensive for the attack from the time perspective of losing the opportunity to attack, and from the financial perspective of cost per sample. A common method to speed up the solution search is applying momentum with gradient descent, however, the presence of noise in stochastic gradients due to channel noise can nullify the theoretical gain of momentum [65]. As a result, it is important to consider the noise in the design of the adversarial sample generation process.

VoIP Speaker Highlight/Attribution: The speaker highlight is a feature in modern VoIP systems to assist in multi-party voice communications. While this is an important feature for usability, it is also a concern for attribution to the attacker. Similar to the channel analysis, based on the reverse engineering result of the VoIP software, the exploration of the attribution mechanism focused on several key attributes of speech volume, frequency, and duration.

Data Collection: To perform the measurement, three computers are used to join a zoom meeting: one is playing a TED talk to simulate a presentation via VoIP, while another is acting as the attacker playing different audio clips. Whether or not the attacker is highlighted is recorded to calculate the attribution rate over 10 repetitions. The TED talk audio is played at an average loudness of 70 dBA, since that is common for daily conversations [12]. Both attacker and victim computers have the same hardware and software stack. We also

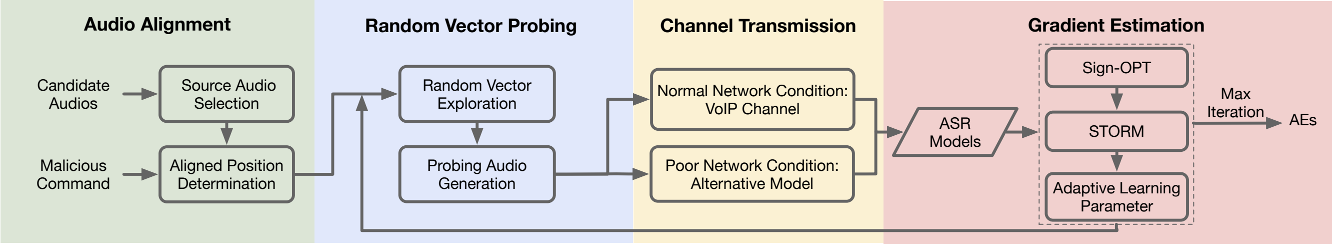


Figure 3: TAINT attack pipelines.

investigated the highlight rate for clips at various frequencies, and found the speaker highlight has little to do with audio frequency.

Measurement and Interpretation: Shown in Figure 2(f) is the attribution rate at different audio lengths and loudness levels. While attribution rate significantly increases as attacker volume gets close to background speech, as long as it stays 10 dBA away, there is little to zero chance of attribution. Furthermore, a shorter audio also reduces the possibility of speaker highlight.

Insights: To avoid attribution, it is important that samples are played at a volume that's lower than the speaker, and furthermore, shorter audio length also mitigates the differences in loudness. Even though our key technical contribution is not on attribution avoidance, our sample generation and audio injection do place an additional constraint on both the volume and length of the AE.

5 TAINT ATTACK DESIGN

5.1 Formulation and Overview

Formulation: For acoustic systems, given a source audio x where $SR(x) = y$, we will craft an adversarial sample y^*, such that

$$y^* = SR(H(x+\delta)), \text{s.t. } \|\delta\| < \eta, y^* \neq y, \tag{1}$$

where H denotes the channel transfer function, SR is the target ASR model. When the attacker has internal knowledge, we can find AE by solving the objective function

$$\text{argmin}_\delta L(SR(H(x+\delta)), y^*) + \alpha \cdot dB_x(\delta), \tag{2}$$

where L is the loss function, α is the weighting factor to limit the perturbation. Without any internal knowledge of SR and H, however, we cannot compute the loss L or its gradient. Instead, we reformulate the problem as

$$\text{argmin}_\delta L(\delta) = \begin{cases} \|\delta\|, & \text{if } SR(H(x+\delta)) = y^*, \\ +\infty, & \text{otherwise.} \end{cases} \tag{3}$$

Overview: An overview of the design is shown in Figure 3. There are four key steps in the baseline pipeline in TAINT. The baseline assumes a working network connection that provides common VoIP as discussed in Section 4. Similar to [69], we first combine the target and source sample, followed by a gradient-estimation-based sample generation process to add perturbations to the combined audio while maintaining the classification of the sample. There are three key design elements in TAINT, 1) using gradient-estimation to overcome impacts of VoIP channels, 2) generation and deployment time imperceptibility measures, and 3) alternative network model for sample re-training during poor network connections.

5.2 Overcoming the VoIP Impacts

Challenges and Problem Formulations: From the preliminary experiments of applying existing techniques to generate AE over VoIP channels in Section 3, it can be observed that the signal noise and distortion over VoIP raise new challenges in navigating the decision space in adversarial sample generation.

There are three key challenges. 1) Upon closer examination of the Zoom VoIP channel in Section 4, we found that the channel often has a nonlinear frequency response coupled with noises from audio encoding/decoding and random network events. Thus, the first challenge is that $H(\cdot)$ is a complex non-linear function. 2) Second, we want to demonstrate the feasibility of attack on different real-world systems. Our goal is then to tackle the more challenging black-box setting where neither the confidence score nor the gradient can be obtained. The second challenge is that we are tackling a hard label black-box setting that requires significantly more queries to probe the decision boundary since there is little information from each query [69]. 3) Both the first and the second challenges demand non-trivial expansion of the number of queries. The third challenge is query efficiency, which we found during the prototype process of our exploration. From the financial perspective, each query has a tangible financial cost. Training a 10s sample using the techniques from [60] takes 300,000 queries even without the VoIP channel, and can cost up to 1,200 USD using Google Cloud Speech-to-Text [9]. Further, commercial APIs also limit the number of queries to avoid denial of service [9]. Repeated queries of similar audio content in a short time also trigger the alarm by the service provider's intrusion detection system [34, 58]. While all these query limitations can be mitigated for more powerful attackers, it does raise the bar from both technical and financial perspectives.

To summarize, from overcoming the VoIP channel perspective, the proposed solution also has to overcome the complex non-convex transfer function due to VoIP channel, and tackle the more restrictive hard label black-box model while maintaining query efficiency.

Our Solution: In this work, we take a different approach towards search exploration of black-box models. Instead of using evolutionary algorithms that are widely adapted in recent [39, 60], we make use of gradient estimation. The key consideration is the ability to converge quickly in a search space with high uncertainty. As discovered in the preliminary work outlined in Table 1, existing approaches fail to converge due to the addition of VoIP channel. One of the key contributing factors is the query efficiency of adaptive evolution algorithms, which provides better ability to search large space in a less greedy manner. To address this challenge, we adapt the gradient estimation approach, and build on top of Sign-Opt, a query-efficient gradient estimation-based boundary attack proposed in the image domain [37], to minimize Eq. 3. The intuition of the algorithm is to generate Q random vectors to probe the decision boundary. The gradients at each step t is estimated by

$$\nabla_{\delta} L_t = \frac{1}{Q} \sum_{q=1}^{Q} \operatorname{sign}(L(H(\delta + \epsilon \mathbf{u}_q)) - L(H(\delta))), \tag{4}$$

$$\begin{aligned} &\operatorname{sign}(L(H(\delta + \epsilon \mathbf{u}_q)) - L(H(\delta))) \\ &= \begin{cases} -1, & \text{if } SR(H(x + L(\delta) \frac{\delta + \epsilon \mathbf{u}}{\|\delta + \epsilon \mathbf{u}\|})) = y^*, \\ +1, & \text{otherwise.} \end{cases} \end{aligned} \tag{5}$$

To obtain a precise gradient estimation, we need a large Q to incorporate random vectors $\mathbf{u}$ with diverse search directions. Yet, this fundamentally conflicts with the goal to limit the number of queries. To address this problem, we update Q adaptively based on its distance to the decision boundary. As the search process approaches the boundary, Q is increased incrementally to strike for a better estimate of the gradient.

However, while the approach above tackles the convergence problem by taking a more greedy approach from optimization perspective, direct application of gradient estimation still suffers from channel noise and distortion. To further adapt to improve the gradient estimation methods, we dive into problem and leverage two insights from channel analysis to tackle the challenge. 1) First, momentum algorithm is one of the most effective tool to accelerate the search process [52], however the presence of noise in the stochastic gradients can nullify the theoretical gain of the momentum algorithm [65]. From our channel analysis in Section 4, it can be observe that there exists a non-trivial amount of noise in the channel even if the network connectivity is great. To address the problem, we leverage Stochastic Recursive Momentum (STORM) algorithm [38] to reduce the variance of the estimated gradients. A more effective estimation of gradient is as follows,

$$\nabla_{\delta} G_t = (1-\alpha) \nabla_{\delta} G_{t-1} + \alpha \nabla_{\delta} L_t + (1-\alpha)(\nabla_{\delta} L_t - \nabla_{\delta} L_{t-1}). \tag{6}$$

2) Second, as shown in the channel analysis, existing VoIP solutions often filter out certain frequency ranges that are less important for voice quality to improve performance. This significantly impact the quality of the adversarial sample, since it can filter our malicious perturbations that are added intentionally to either change the decision or to make the sample more imperceptible, and therefore has to be taken into consideration in the sample generation process. To mitigate this, a band-pass filter is applied to limit the frequency range of perturbations and the perturbations are updated by

$$\sigma_{t+1} = \sigma_t - \eta_t f_{BPF}(\nabla_{\delta} G_t), \tag{7}$$

where α is the momentum weight, η is the learning rate, f_{BPF} is the band-pass filtering function. To ensure a fast convergence rate, the α and η will be updated according to the past gradients [38] by

Algorithm 1 TAINT Attack Algorithm

```
Require: Source audio $x$, initial command audio $c$, maximum queries $c_{max}$,
    standard deviation $\delta$, weight parameter $\alpha, \beta, \eta$.
Ensure: Adversarial samples $x^*$, best SNR $S_{max}$.
  Generate initial $x^*$ by aligning $\sigma$ into $c$ following the position given in
  Eq. 11, 12, 13;
  $\sigma = x^* - x$, $\theta = \sigma / \|\sigma\|$;
  Update $\|\sigma\|$ via binary search algorithm by $a$ queries;
  $c = 0, t = 0$;
  $S_{max} = SNR(\|\sigma\| \times \theta)$, $c = c + a$;
  while $c < c_{max}$ do
      Initialize play audio list $\mathbf{p}$;
      for $i = 1$ to $Q$ do
          $\theta_i = \theta$;
          Sample $u_i \sim \mathcal{N}(0, \delta^2)$ ;
          $u_i = u_i / \|u_i\|$;
          $\theta_i = \theta_i + \beta * u_i$, $\theta_i = \theta_i / \|\theta_i\|$;
          Append audio $p_i = x + \|\sigma\| * \theta_i$ into $\mathbf{p}$;
      end for
      Play $\mathbf{p}$ for $m$ times and get decisions;
      Compute gradient of the loss function $\nabla_{\delta} L_t^{'}$ by Eq. 4, 5;
      Compute the step gradient $\nabla_{\delta} G_t^{'}$ by Eq. 6;
      Update $\theta$ by Eq. 7;
      Update $\|\sigma\|$ with $\theta$ via binary search by $m$ queries;
      Update $\alpha$ and $\eta$ by Eq. 8, 9;
      if $SNR(\|\sigma\| \times \theta) > S_{max}$ then
          $x^* = \|\sigma\| \times \theta$;
          $S_{max} = SNR(x^*)$;
      end if
      $c = c + Q + m$, $t = t + 1$;
  end while
  return $x^*$, $S_{max}$
```

$$\eta_t = \frac{k}{(w + \sum_{i=1}^{t} \|\nabla_{\delta} L_i\|^2)^{1/3}}, \tag{8}$$

$$\alpha_{t+1} = c \eta_t^2, \tag{9}$$

where k,w, and c are tuned parameters. More details of the algorithm are given in Alg. 1. Similar to existing work, a key goal of the optimization besides successful transcribing to target phrases is the signal-to-noise ratio (SNR), which is calculated by computing the ratio of the average power between signal and noise. In our experiment, we set initial learning rate as $\eta = 0.05$ and probing size $\beta = 0.001$. The algorithm goes through c_{max} iterations and return the adversarial samples with the best SNR.

5.3 Tackling Imperceptibility and Cyber/Physical Interference

Challenges: Imperceptibility is one of the most important attributes of adversarial samples. When adversarial audio sample is delivered via VoIP video conferencing software, there are additional challenges this work has to tackle. More specifically, 1) The design decision to search for solution space greedily using gradient-estimation with recursive momentum may have a negative impact on the quality of the solution. In our approach, the quality translate to SNR, which is related to imperceptibility of the generated sample. However, SNR is only part of the solution, we leverage psychoacoustic hiding to complement the widely adapted SNR-guided optimization. 2) In modern VoIP conferencing system, the speaker highlight feature highlight attributes voice to a user, potentially exposing the attacker. In order to avoid attribution, there are additional constraints on the volume and the length of the adversarial audio based on our channel analysis in Section 4. Furthermore, since everyone is listening in the VoIP channel, it limits the source audio one can use, since it has to be an organic household noise you would expect in a video conferencing, such as car horn, policy siren, road side/airplane noise, baby crying, etc. 3) Lastly, there could be interference from both target victim's physical environment (folks talking in the background or TV) and the VoIP's (folks talking or giving a talk over VoIP conferencing software). Similar to other adversarial audio attacks, the long standing challenge of background interference also applies to our work. This issue is also exacerbated by the diverse sources of interference.

Improving Imperceptibility: Efforts in this direction generally fall into two stages, sample generation and deployment.

Psychoacoustic Hiding in Sample Generation: Psychoacoustic hiding principles refer to the phenomenon that a louder signal can make signals at nearby frequencies (frequency masking) and time (temporal masking) imperceptible [47]. It was adapted in [54] to make adversarial samples imperceptible to human during over-the-air attack. Instead of applying these methods to the final AE, we propose to consider this during source and target audio merging process. Source audio selection and audio alignment algorithm are two key design elements in this process. Given the short-time Fourier transform (STFT) of the target command audio x, the log-magnitude power spectral density (PSD) can be calculated as

$$p_x(k) = 10\log_{10}\left|\frac{1}{N}s_x(k)\right|^2, \tag{10}$$

where $s_x(k)$ as the kth bin of the STFT of audio x. Then the normalized PSD estimate $\hat{p_x}(k)$ can be calcualted following [47] by

$$\hat{p_x}(k) = 96 - \max_k p_x(k) + p_x(k). \tag{11}$$

Next, the masking threshold $\theta_s(k)$ of source audio s can be calculated in three steps. First, frequency maskers are identified, then each masker's masking threshold is approximated using a two-slope spread function. Lastly, the global masking threshold is obtained by combining the individual masking threshold via logarithmic domain addition. Suppose we have a source audio list S, first, we calculate the best-aligned position in each source audio s_i by

$$t_i = \underset{t^*}{\operatorname{argmin}} \sum_{t=t^*}^{t^*+t_0} \sum_k \max(0, \hat{p_x}(t,k) - \theta_{s_i}(t,k)), \tag{12}$$

Then, we choose the source audio by

$$s^* = \underset{s_i}{\operatorname{argmin}} \sum_{t=t_i}^{t_i+t_0} \sum_k \max(0, \hat{p_x}(t,k) - \theta_{s_i}(t,k)). \tag{13}$$

where t_0 is the duration of target command audio. The selection criteria of good source audio are that most of the masking threshold is higher than $\hat{p_x}(k)$ in a continuous-time and frequency range.

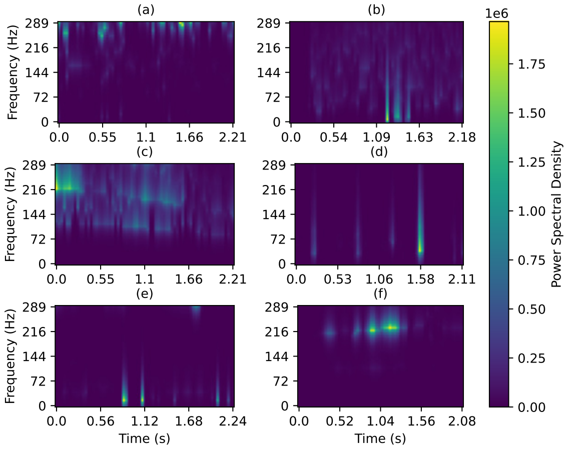


Figure 4: The spectrogram of the psychoacoustic hiding thresholds of the different source audios.

Concrete cases are shown in Fig. 4, we list the spectrogram of the psychoacoustic hiding thresholds of six common source audios, that can often be heard during in meetings. Airplane sound (Figure 4 (c)) and siren sound (Figure 4 (b)) are the most suitable selections for source audios since both of them have a large continuous time-frequency space to hide the target command audio. Although the car horn (Figure 4 (a)) also contains a continuous region, it is concentrated in the high-frequency band, which is not effective in hiding the whole spectrum of malicious commands. The keyboard typing (Figure 4 (d)), child speaking (Figure 4 (e)), and bird chirping (Figure 4 (f)) are also not good fits for the source audios as they only contain discrete hiding regions.

Deploy Time Mitigation: At deploy time, the adversarial sample needs to be delivered at the appropriate time and loudness to strike a delicate balance between perceptibly/attribution rate and attack success rate. The environmental interference is a known open challenge, and the addition of speaker highlight also makes the deployment of sample challenging. While this is not the primary focus of our paper, there are several mitigation mechanisms we make use of in the deployment to make the attack practical. 1) To mitigate the attribution via the speaker highlight in VoIP conferences, it is important to keep the adversarial voice short and several dBA below the current active speaker. More specifically, we tune the speech speed in the adversarial audio such that the sample duration is minimized yet still recognizable, we also manually tune the loudness of the clip based on the past history of the zoom active speaker volume to avoid speaker highlight. 2) To mitigate the interference from other speakers either in cyber (VoIP) environment or the physical (people talking in the victim's room) environment, we make multiple attempts in the adversarial audio, this requires a special selection of source (carrier) audio to be organically repetitive types, such as renovation noise, air dryer noise or baby crying to make it less alarming. However, these samples may not be ideal for acoustic hiding depending how well they are spread out in the frequency spectrum, therefore source audio has to be carefully selected. 3) Lastly, given speech interference is simply the superposition of

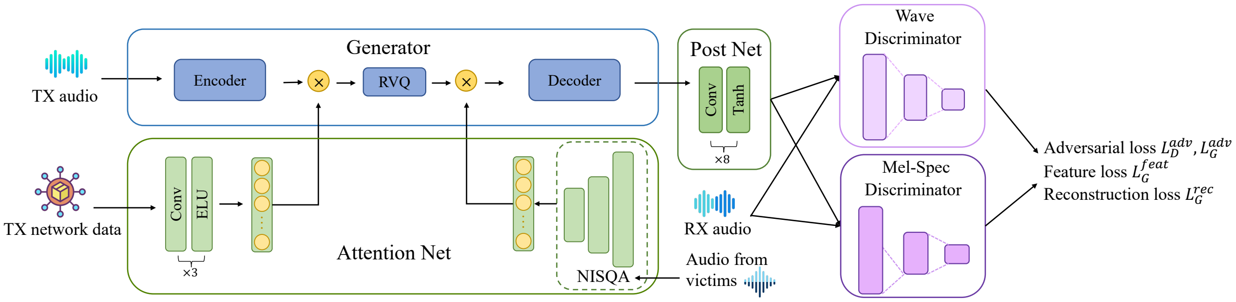


Figure 5: Overall architecture of the proposed alternative model for VoIP channel.

the audio signals of the AE and the interfering audio at the ASR. Under some circumstances, it may be possible to anticipate what that signal is. For example, many VoIP conferences start with "Good Morning" and end with "Thank you". In these cases, it is possible to train the sample anticipating the interference source to significantly improve the probability of success of the AE. While these three mitigation are immensely effective in boosting our attack success rate while avoiding attribution and tolerating the audio interference, much of the mitigation is currently manual tuning and labor-intensive. In the future, we will explore mechanisms to automate these techniques to tackle the challenges.

5.4 Alternative Model for Poor Net Connectivity

Challenge and Problem Formulation: Another practical challenge lies in the varying network conditions that could severely impact the delivery of adversarial examples. For example, the victim may make a video call in a cafe with poor network connections, and he is likely to suffer from high packet loss and delays. In such scenarios, directly applying adversarial examples pre-trained on a different network condition is likely to fail. On the other hand, it is almost impossible to pre-train all the adversarial examples under various network conditions by enumerating all potential combinations of network parameters. As a result, the wide variety of poor network conditions pose additional challenge to our attack.

Our Solution: To tackle this challenge, we propose an alternative model to approximate the impacts of the targeted VoIP channel under different network conditions. Designing an effective alternative model for VoIP channels is non-trivial and has several challenges. First, modern VoIP services generally incorporate multiple processing functions to preserve communication quality, such as noise suppression, echo cancellation, and codec. How to effectively model a combination of these functionalities becomes a challenge. Besides, audio transmitted over VoIP is generally sampled in high frequency (e.g., 16 kHz), which makes the audio on both the transmitter (i.e., attacker) and receiver (i.e., victim) sides long sequences by nature. As such, it becomes difficult to build an effective sequence-to-sequence model. Therefore, we borrowed insights from existing work on modeling VoIP channels and designed a neural-network-based model.

As shown in Figure 5, the proposed model consists of five components, namely generator, post net, wave discriminator, mel-spectrum discriminator, and attention net. Our model adapts generative adversarial networks (GANs) [44] design, where the generator produces audio samples while discriminator is incorporated to distinguish the predicted audio from ground truth. We adopt the same architecture of GAN as [67]. The output of the generator is then processed by post net to improve the audio quality, which consists of 8 1D-convolutional layers and a Tanh layer [57]. We further leverage the attention net to emulate impacts on the VoIP channel from the network connectivity, where the attention module is utilized to select key network features. NISQA [50] is applied to extract network features from the audio on the victims' side. To facilitate generation of samples that balances audio fidelity and perceptual quality, the loss function is designed to incorporate adversarial loss (L_G^{adv}), feature loss (L_G^{feat}), and spectral reconstruction loss (L_G^{rec}) to trade-off between perceptual quality and distortion magnitude (i.e. fidelity), as shown in Eq. 14.

$$L_G = \alpha L_G^{adv} + \beta L_G^{feat} + \eta L_G^{rec}. \tag{14}$$

The adversarial loss is designed to improve the perceptual quality, and we adopt 3 wave discriminators and 1 STFT discriminator [67]. The feature loss is designed to improve the fidelity, and a multi-scale reconstruction loss [40] is applied to further enhance the fidelity of the predicted audio [67].

6 EXPERIMENT

The evaluation of TAINT starts with the baseline condition where there is neither network congestion nor environmental interference, then progresses to more challenging environments such as poor network connectivity, realistic household, or Zoom environment. Lastly, we also employed an IRB approved user study to evaluate the imperceptibility of the generated adversarial samples.

6.1 Experiment Settings

Audio and Speech Datasets: To simulate real meeting scenario, (1) we selected source environment sound that would appear in a daily meeting from a widely-used FSD50K dataset [42] such as sirens, road noise, car horns, bird chirping, keyboard typing, etc. (2) We chose ten frequently-used intelligent device control commands as malicious commands[2] [3], and synthesized them with text-to-speech

[2] The malicious command includes open the website, navigate to my home, turn off the light, airplane mode on, login PayPal, call my wife, play scary music, send a text, open the door and activation word for each ASR (OK Google, Hey Cortana, Echo). In IBM, we replace the activation word with turn on the gas stove.

[3] As Amazon Echo does not response to certain commands above, we further choose the command for Amazon Transcribe/Echo: call 911, clear notification, turn on the TV.

services. The datasets used to pre-train the alternative model is TIMIT Speech Corpus [43], we played each speech at TX over Zoom and recorded the audio in RX. We also played the speech in RX and recorded the audio in TX under the same network condition to simulate the victims' speech. To simulate different network conditions, we used *tc* [22] and *Trickle* [23] to generate various network traffic with different network delays, packet loss rates, and bandwidths. During the audio recording, we also recorded the network monitoring window with zoom provided timestamp to measure the network status between TX and the Zoom cloud server, and utilized Tesseract OCR [56] to extract text from the recorded video. To collect the fine-tuned datasets, we ran TAINT algorithm on target commands to generate AEs with different levels of SNR, then used the same methods as above to collect the datasets of AEs. During our experiment, we selected two phrases with 2000 samples each.

Video Conferencing Software: We chose the five most popular VCS platforms [1]: Zoom (version: 5.9.6) [26], Microsoft Teams (version: 1.5.00.9159) [16], Skype (version: 8.83.0.408) [21], Webex (version: 42.4.0.21893) [25], and Google Meet (version: 87.0.0) [10] as our attack mediums.

Target ASR System: To evaluate the performance of our attacks, we chose the representative commercial cloud speech to text (STT) services: Google Cloud Speech-to-Text (Google STT) [8], Microsoft Azure Speech to Text (Microsoft STT) [14], Amazon Transcribe (Amazon STT) [5], and IBM Speech to Text (IBM STT) [13]. We also chose the corresponding IVC devices: Google Assistant (Ver. 1.9.40904) [7], Microsoft Cortana (Ver. 4.2204.13303.0) [15], and Amazon Echo (3rd Gen) [4]. IBM does not have its own IVC devices. We only used the final decision provided by the ASRs.

Hardware: The two speakers we used for over-the-air experiment were Philips HTL 1508/37 Sound bar and JBL Pulse 2 portable speaker. The Google Assistant App ran on an iPhone 12 Pro Max, Microsoft Cortana ran on an HP OMEN 15-ax019tx laptop. The AEs on IBM ASR were recorded by an iPad Pro (5th gen). We used a digital sound level meter BAFX3370 to measure the volume.

Evaluation Metrics: We used the attack success rate (SR) to evaluate the effectiveness of AEs. Success rate is defined as the proportion of adversarial examples that can successfully attack target systems among all tested samples. In over-the-air attack evaluation, we recorded success rates within multiple attempts: one attempt (1x), two attempts (2x), and three attempts (3x). We used signal-to-noise ratio (SNR) to describe the perturbation on audio AEs and the number of queries on the target model to indicate the efficiency of the attacks. To evaluate the efficiency of alternative model, we tested the time overhead including both the model inference time and the total time cost. In addition, to show the imperceptibility of our AEs, we conducted a user study to analyze human perception.

6.2 Baseline Normal Network Evaluation

Over-the-line Attacks on Cloud APIs: We evaluated our attack against several voice transcription services over the line to prevent or mislead mass surveillance as in [28]. Table 2 shows the performance of TAINT attacks on different commercial speech to text APIs after 1500 queries. We pre-trained and tested AEs in the normal network with up/downlink bandwidth greater than 100Mbps. It took 1500 queries to train the samples. To avoid bias, different source audios were used for different targets, including siren, road noise, airplane noise, etc. We also made an effort to ensure consistency in audio length on SNR. It can be observed that the attack works across all existing speech-to-text systems, with relatively high SNR (one of the metrics for imperceptibility).

Table 2: Attack success rate (SR) and SNR of over-the-line attacks on speech-to-text APIs over VCSs.

VCS	Google STT		Microsoft STT		Amazon STT		IBM STT	
	SR	SNR (dB)	SR	SNR (dB)	SR	SNR (dB)	SR	SNR (dB)
Zoom	10/10	16.01	10/10	13.54	10/10	13.53	10/10	11.13
Teams	10/10	16.30	10/10	16.81	10/10	12.50	10/10	9.93
Skype	10/10	18.35	10/10	13.23	10/10	13.94	10/10	10.99
Webex	10/10	15.91	10/10	13.02	10/10	13.95	10/10	10.13
Meet	10/10	16.90	10/10	15.47	10/10	12.83	10/10	10.23

Note that, we use a maximum 1500 queries to craft each AE.

Over-the-air Attacks on Voice Assistant Devices: As shown in [36], the adversarial samples are transferable to the IVC devices from the API services in the same company. Thus, we first crafted samples over commercial Speech-to-Text APIs, and then transfered the adversarial samples to attack the corresponding IVC devices. To give a more comprehensive evaluation, we followed different definitions of attack success rate: the audio AEs can be correctly recognized by the devices as the target command within single attempt (1x), two attempts (2x), three attempts (3x). The experiment was conducted in a meeting room (3.7 meter long, 2.9 meter wide, and 3 meter tall), we used Philips HTL 1508/37 Soundbar as the speaker to play AEs, and the device was placed within 0.5m around the speaker. As shown in Table. 3, we have achieved a high success rate and SNR in all the IVC devices among different VCSs. For example, we have achieved 100% success rate in an average SNR of 12.94 over Zoom and air. Although SNR is lower when compared to pure attack over Zoom to API, both success rate and SNR are higher when compared to the state-of-the-art methods OCCAM and Devil's Whisper (shown in Table 1). The possible reason behind this could be - AEs that could survive over a VoIP network were more robust, consequently surviving over the air better. Another interesting finding is that the success rate is quite stable among different attempts although more attempts would lead to a higher success rate. This is likely due to the fact that the level of over-the-air interference is low when the distance is small. As shown in later experiments, the attack became more unstable when the distance increases. Also, the attacks against Google Assistant over Microsoft Teams only achieved a success rate of fifty percent. A potential explanation to the result might be the source audio we chose, an airplane sound, can possibly fail to be recognized by the voice activity detection (VAD) module in Google Assistant.

6.3 Impact of Poor Network Connection

Sample Robustness in Different Network Connections: We chose five settings with varying network bandwidth, latency, and packet loss rate to evaluate the performance of over-the-line and

Table 3: Attack success rate (SR) and SNR of over-the-line and over-the-air attacks against IVC Devices over different VCSs.

VCS	Google Assistant				Microsoft Cortana				Amazon Echo				IBM ASR			
	1x SR	2x SR	3x SR	SNR (dB)	1x SR	2x SR	3x SR	SNR (dB)	1x SR	2x SR	3x SR	SNR (dB)	1x SR	2x SR	3x SR	SNR (dB)
Zoom	8/10	10/10	10/10	12.94	8/10	9/10	10/10	12.28	8/10	9/10	10/10	13.11	9/10	9/10	9/10	10.40
Teams	4/10	5/10	5/10	13.83	8/10	9/10	10/10	12.25	8/10	9/10	10/10	12.73	9/10	9/10	9/10	9.46
Skype	7/10	8/10	8/10	15.31	10/10	10/10	10/10	13.28	8/10	9/10	10/10	14.34	9/10	9/10	9/10	10.31
Webex	8/10	9/10	10/10	12.88	8/10	9/10	10/10	11.30	7/10	9/10	10/10	10.34	10/10	10/10	10/10	8.31
Meet	9/10	9/10	10/10	14.02	6/10	7/10	9/10	13.63	8/10	9/10	10/10	13.35	9/10	9/10	9/10	8.34

Note that, (i) ∗x SR represents the success rate after ∗ attemps. (ii) Since we send recorded audios to IBM ASR for transcription, the 1x SR, 2x SR, and 3x SR are the same.

Table 4: Attack success rate (SR) and SNR of over-the-line and over-the-air attacks against Microsoft STT and Microsoft Cortana in different network conditions.

BandW	Lat	Loss	MS STT		MS Cortana			
			SR	SNR	1x SR	2x SR	3x SR	SNR
10 Mbps	50 ms	0%	10/10	13.24	8/10	8/10	10/10	12.08
2 Mbps	70 ms	0%	10/10	13.20	8/10	9/10	10/10	11.92
0.8 Mbps	170 ms	2%	10/10	11.66	8/10	8/10	10/10	11.02
0.6 Mbps	270 ms	4%	9/10	10.51	7/10	7/10	9/10	10.48
0.4 Mbps	320 ms	6%	5/10	9.83	3/10	3/10	5/10	9.44

Note that, (i) BandW means bandwidth, Lat means latency, and Loss means packet loss.

Table 5: Experimental results of TAINT attacks with alternative model on Google speech-to-text API over Zoom in the poor network connectivity.

Command	Query	IF Time‡ (s)	To Time† (s)	SNR (dB)
OK Google	1071	0.78	362.18	18.63
Call my wife	1257	0.87	437.76	17.57

Note that, (i) ‡: "IF Time" means the total inference time of the alternative model. (ii) †: "To Time" means the total time to craft AEs.

over-the-air TAINT attacks under different network conditions. Microsoft Azure Speech-to-Text and Microsoft Cortana are chosen as targets for over-the-line and over-the-air attacks, respectively. Moreover, Zoom served as the VCS for the two kinds of attacks. The experimental settings and results are shown in Table 4.

For both over-the-air and over-the-line attacks, we could achieve a 100% success rate under more than 800 Kbps network bandwidth, less than 170 ms latency, and less than 2% packet loss network condition. When the network condition was worse (600Kbps bandwidth), we could still achieve 90% success rate with average SNR of 10.51 and 10.48 for the two kinds of attacks. However, when the network bandwidth was 400Kbps with 320 ms latency and 6% packet loss, we could only get 50% success rate with SNR 9.83 and 9.44 respectively. Hence the robustness of the samples is not reliable. Samples for the specific degraded network channels have to be retrained.

Benefit of Network Channel Alternative Model under Poor Network Conditions: We conducted the experiment on GeForce RTX 3070 Ti GPU with 8 GB GDDR6 memory. We first pre-trained the alternative model on collected TIMIT datasets with aligned network data for 100 epochs using Adam optimizer with a learning rate of 10^{-4}. Then we fine-tuned the model on AE datasets for 50 epochs with a learning rate of 10^{-5}. We measured the effectiveness of our methods by generating two AEs that failed when using TAINT directly under 1Gbps uplink bandwidth in TX and 400Kbps downlink bandwidth in RX. As shown in Table 5, we only needed less than 1 second to pass through the channel for the whole AE generation process after replacing the channel with the alternative model. Without the alternative model, each query would take the total duration of AEs (usually 2∼3s depending on the length of the audio). Also, it only took about 6∼7 minutes to complete. Most of the time was taken in querying Google Cloud Speech-to-text API (average query latency is about 0.3s according to our observation).

6.4 Diverse Over-the-air System Settings

To validate the robustness of our attacks in different environments, we evaluated our attack in various over-the-air settings (different room layouts, attack distances, and speakers). Specifically, we conducted experiments in two rooms: a meeting room (3.7 meters long, 2.9 meters wide, and 3 meters tall) and a bedroom (4.1 meters long, 3.7 meters wide, and 2.8 meters tall). The attack distances are set as 0.5m, 1m, and 1.5m. The AEs are playing in 60~70 dBA, and the ambient noises are in 50 dBA. The experimental results are shown in Table 6. At a close distance (i.e., 0.5m), our attack could achieve a nearly 100% success rate and a high SNR with different speakers and room layouts within two attempts. The high success rate and SNR could still be maintained at the moderate attack distance (i.e., 1m) with more attempts. The success rate drops at the further distance (i.e., 1.5m). Even though more attempts can increase the success rate, it could also increase the perceptibility of the attack. It is worth noting that the attack does become more unstable as the distance increases. This is because over-the-air interference would become unpredictable as the distance increase [35].

6.5 Evaluation in Meeting and Household Environment While Avoiding Attribution

Background interference in the adversarial sample is an open research problem that practical attacks have to address. We aim to understand the impacts of this interference and the effectiveness of our mitigation in our evaluation. To avoid bias towards our own solution, we specifically selected a volume that is lower than the background noise, since using a very high volume to overpower the noise is known to work, but quite alarming for practical attacks. We evaluated two types of noises, an active participant speaking in Zoom from the cyber domain, and a TV playing and people speaking in the target environment from the physical domain.

Table 6: Over-the-air attack success rate (SR) and SNR against Google Assistant over Zoom in different household settings.

Room Layout	Distance	0.5m				1m				1.5m			
	Speaker	1x SR	2x SR	3x SR	SNR (dB)	1x SR	2x SR	3x SR	SNR (dB)	1x SR	2x SR	3x SR	SNR (dB)
Meeting Room	Philips HTL 1508/37 Sound Bar	8/10	10/10	10/10	12.94	6/10	8/10	9/10	13.25	3/10	7/10	8/10	12.03
	JBL Pulse 2 Portable Speaker	7/10	9/10	10/10	14.19	4/10	8/10	10/10	13.03	4/10	8/10	8/10	12.52
Bedroom	Philips HTL 1508/37 Sound Bar	10/10	10/10	10/10	14.16	8/10	10/10	10/10	12.49	5/10	6/10	6/10	13.49
	JBL Pulse 2 Portable Speaker	7/10	9/10	10/10	11.62	5/10	7/10	9/10	12.19	6/10	6/10	7/10	12.40

Table 7: Over-the-air attack success rate (SR) and SNR against Google Assistant over Zoom while avoiding attribution.

Source	Methods	1x SR	2X SR	3x SR	10x SR	SNR
Zoom	Robust	1/10	2/10	3/10	10/10	6.33
Talking	Mix-Train	3/10	6/10	9/10	10/10	6.86
Talking	Robust	1/10	2/10	2/10	10/10	6.38
TV	Robust	1/10	2/10	3/10	10/10	6.11

Participant Speaking in VoIP Meeting: The experiment is conducted in a meeting room as the victim environment with JBL speakers as the victim's output device at a 0.5m attack distance to the ASR. We chose Google Assistant as the target and Zoom as the VCS. The active talking level is 68~73 dBA, which is simulated by continuously playing a TED Talk video on one meeting participant's side. The experiments were carried out 10 times at different random onsets of the TED talk to avoid bias on interference. For each attack, ten AEs were used to avoid bias in samples. The experimental results are given in Table 7. To avoid speaker focus, we had to lower the volume and shorten the audio length of AEs according to the analysis in Section 4. Consistent with other results in the field [35], even after manually tuning the adversarial voice to be as close as to the active speaker's volume (varying among 62~75dBA) while shortening the length to squeeze several more dBA, the interference still significantly diminished the attack success rate to only 10% on the first attempt. However, the success rate did gradually increase linearly as we make more attempts. One of the mitigations we propose is to repeat the attack. Since attribution can be avoided, it is possible to continue the attack while remaining stealthy. We found that the attack could succeed after 10 attempts almost all the time. A key observation on this is that due to the dynamics of the interference, making more attempts allows the same AE to be interfered with differently, thus contributing to the improvement of the success rate. We also evaluated the alternative method of leveraging anticipated interference to our advantage. In some cases, the attacker may be able to anticipate what the speaker would say, such as "Morning", then it becomes possible to develop an AE that will superimpose on the interference to become the adversarial audio that can cause the attacker intended transcription. We labeled this as mix-training, and the success rate was much higher at 90% while avoiding the attribution.

Household Environment: We evaluated two practical scenarios: 1) TV is playing some programs or shows; 2) Someone is talking in the background in the same physical place with victims who are having a meeting. The experiment is conducted in a bedroom with a victim JBL speakers at a 0.5m attack distance to the ASR. We used TED Talk to simulate TV playing, the level is between 68~73 dBA. To simulate people talking, we recruited one volunteer to continuously read a news report from The New York Times and try to locate his voice between 68~73 dBA to be consistent. The results are similar to the meeting interference, as shown in Table 7. We did not conduct any experiment on mix-train for this environment, because we believe it is impractical to be able to anticipate household conversations as an attacker. He/she would need to rely on making more attempts.

Table 8: Human perception evaluation on TAINT Attacks.

VCS	Normal (%)	Intrusive (%)	Talking (%)	Recognize (%)			
				1x	2x	3x	5x
Zoom	49.5	30.5	20.0	0	0	0	0
Teams	69.4	24.2	6.4	0	0	0	0
Skype	32.9	18.7	48.4	0	0	1.1	1.1
Webex	50.1	37.0	12.9	0	0	0	0
Meet	49.0	41.3	9.7	0	0	1.1	1.1

Note that, (i) "Normal" refers to the participant recognizes the audio as normal/organic environmental noise. (ii) "Intrusive" refers to the participant recognizes the audio as intrusive alarming noise. (iii) "Talking" refers to the participant recognize someone talking in the background. If the participant recognizes people talking in the audio, he/she is then asked to identify the content of such talking. (iv) ∗x represent the percentage of users that can correctly identify the content of the AEs in ∗ attempts.

6.6 Imperceptibility User Study

Although SNR can provide us with an intuitive and rough estimation of the AE's stealthiness [36], it still cannot capture the full perception of humans. To this end, we followed up with a user study, which is approved by the Institutional Review Board (IRB) and classified as exempt. Our study is designed to avoid any potential (psychological, social, legal, physical, etc.) risks to the participants. Following the existing survey methodology for user studies [36, 69], our survey does not collect any confidential information on the participants, only the demographic data is collected. The data will be securely deleted upon the completion of the study. Furthermore, in an effort to minimize discomfort for our participants, we had selected household noises such as police sirens and highway noises as source audios. The adversarial speeches were also selected to be household common phrases, such as "OK Google, call my wife". To avoid any damage to hearing, we also normalize the volume.

One working AE was randomly selected for each of the VoIP platforms (Zoom, Microsoft Teams, Skype, Google Meet, and Cisco Webex) for the user survey. Each participant had to listen to each audio clip at least three times without an upper limit. For each audio, they were asked to choose among three options, "the audio is fine", "the audio has a lot of noise", and "the audio has someone talking in the background". They were also also asked to put down how many times each audio was listened to,helping us examine the perceptibility of the AEs. However, since source audios for our attack scenario are generally daily noises, we also specifically asked the participants to describe the noise. This allows the separation between organic environmental noise and intrusive alarming noise. Using this response, we separated the responses into the category of "normal/organic environmental noise" and "intrusive noise".

We received 95 replies from the U.S., the U.K., China, Saudi Arabia, etc. There are 33 males and 61 females, and one prefers not to identify. 6 of the participants of 18 years old, 29 are within 22 to 25 years old, 16 are within 26-29 years old, and 44 are above 30 years old. 88.42% of participants using video conference software in their daily life, among them, 86.90% use Zoom, 39.29% use Microsoft Teams, 21.43% use Skype, 14.29% use Webex, 8.33% use Google Meet, 11.90% use others. All participants speak and understand English. For the non-native speakers that account for 62.1% of all the participants, most of them (91.5%) have educational background of bachelor and above. Table 8 shows the results of our user study. While many participants consider what they hear as noise, they were identifying the noise as siren and noise, which is consistent with the source audio. One notable result is that even after several attempts, participants are unable to identify the concrete contents.

7 DISCUSSIONS

Defense via Audio Downsampling: According to the Nyquist's theorem, the downsampling operation will eliminate high-frequency components from the original audio. Therefore, applying downsampling to the audio inputs can serve as a potential defense as it disrupts well-crafted adversarial examples in the high-frequency domain at the cost of degrading audio quality. In the context of our attack delivered over VoIP, such downsampling operations can be incorporated and applied during transmission, and the original audio will be recovered via upsampling. For example, Zoom uses a sampling frequency of 48KHz. Ten AEs for Google Cloud Speech-to-text were first downsampled to 32KHz, then upsampled back to 48KHz, this reduce the attack success rate to only 30%. If downsampled to 16KHz, none of the AEs could be recognized correctly by both over-the-line and over-the-air at a 0.5m attack distance. However, under the strong assumption that when the adversary can guess the downsampling rate (such as using the network-based side channel), it may be possible to further optimize the attack, partially bypassing the defense, by crafting the AE using the downsampling rate. Using the same source audio, we retrain the ten AEs at 16KHz before upsampling them to 48KHz, then go through downsampling and upsampling steps again. The resulting AEs have 90% success rate in both over-the-line and over-the-air at the attack distance of 0.5m. In a weaker assumption where the attacker can estimate a close but inaccurate sampling rate such as 20KHz, training the samples at this estimated sampling rate and repeating the above procedures, the AEs can achieve a 40% success rate for both over-the-line and over-the-air at 0.5m attack distance. In practice, it can be quite difficult to know the exact downsampling rate.

Defense via Adversarial Training: Besides deploying defenses in video conferencing software, another potential defense lies in the enhancement of the ASR system itself. Adversarial training is one of the most effective approaches to defend against adversarial examples in both image and audio domains [31, 45].

Limitations on Acoustic Interference in Target Cyber/Physical Environment: The need to balance attribution avoidance, i.e. imperceptibility and various noisy target environment is one of the difficult trade-offs in the proposed system. Particularly, as shown in our evaluation, speech in the target environment as well as in the meeting can significantly degrade the adversarial sample performance. Even though this is an open challenge that generally applies to existing literature of adversarial voice [27], the coupling with the fact that our over-the-air attack has to be delivered via the speaker of the victim's PC does make it challenging, and often takes multiple attempts to succeed. To remain out of focus from the active speaker view, the attacker has to transmit AEs with a volume that's slightly lower than the active speaker. If timed correctly, such as injecting the sample when the active speaker is playing music or taking a pause, it is often possible to mitigate the background noise. Furthermore, leveraging anticipated interference is shown to be promising in our preliminary experiment.

Limitation in Network Model: When there is a change in the VoIP signal processing pipeline, alternative models and AEs also have to be updated. One possible solution is to apply incremental learning [61], which updates existing models' weights incrementally according to the new training data. There is also the need to implement a better mechanism to estimate the VoIP channel quality, which we simply assume to be symmetric in our current work. However, this is a well-studied problem in computer networks [30].

Limitations on Participants Recruitment: In our survey study, the participants were recruited across different regions to cover a wide and diverse population. While all the participants understand and speak English fluently, 62.1% of the participants are not native English speakers. This could affect their perception of the adversarial audio since they may be less sensitive to English phrases.

8 CONCLUSIONS

In this paper, we proposed TAINT, the first targeted, query-efficient, black-box, adversarial attack on commercial speech recognition platforms over VoIP network. To address the challenges caused by the unique channel characteristics of VoIP such as frequency selectivity, signal distortions, and random noises, we propose noise-resilient efficient gradient estimation methods to ensure a steady and fast convergence of the adversarial samples. We demonstrate our attack in both over-the-air and over-the-line settings, on four commercial automatic speech recognition systems over five most popular VoIP conferencing software.

ACKNOWLEDGMENT

We thank the anonymous reviewers for their constructive feedback, Ao Li and Shixuan Zhai for their help on the reverse engineering efforts in this project. This work is supported in part by US National Science Foundation under grants CNS-1916926, IIS-1905558, ECCS-2020289, CNS-2038995, and CNS-2154930, and by Army Research Office under contract W911NF-20-1-0141, W911NF-19-1-0241, and W911NF-18-1-0208.

REFERENCES

[1] 84 current video conferencing statistics for the 2021 market. https://www.trustradius.com/vendor-blog/web-conferencing-statistics-trends.

[2] 90-day security plan progress report: April 22. https://blog.zoom.us/90-day-security-plan-progress-report-april-22/.

[3] Alexa skills. https://developer.amazon.com/en-US/alexa/alexa-skills-kit. Accessed: 2022-07-25.

[4] Amazon echo. https://www.amazon.com/echo-3rd-generation/s?k=echo+3rd+generation. Accessed: 2022-07-25.

[5] Amazon transcribe. https://aws.amazon.com/transcribe/. Accessed: 2022-07-25.

[6] Every company going remote permanently: Apr 28, 2022 update. https://buildremote.co/companies/companies-going-remote-permanently/. Accessed: 2022-03-30.

[7] Google assistant. https://assistant.google.com/. Accessed: 2022-03-30.

[8] Google cloud speech-to-text. https://cloud.google.com/speech-to-text/. Accessed: 2022-07-25.

[9] Google cloud speech-to-text price. https://cloud.google.com/speech-to-text/pricing. Accessed: 2022-07-25.

[10] Google meet. https://meet.google.com/. Accessed: 2022-07-25.

[11] How google meet supports two million new users each day. https://cloud.google.com/blog/products/g-suite/how-google-meet-supports-two-million-new-users-each-day.

[12] How many decibels does a human speak normally. https://decibelpro.app/blog/how-many-decibels-does-a-human-speak-normally/. Accessed: 2022-07-25.

[13] Ibm speech to text. https://www.ibm.com/watson/services/speech-to-text/. Accessed: 2022-07-25.

[14] Microsoft azure speech to text. https://azure.microsoft.com/en-us/services/cognitive-services/speech-to-text/. Accessed: 2022-07-25.

[15] Microsoft cortana. https://www.microsoft.com/en-us/cortana/. Accessed: 2022-07-25.

[16] Microsoft teams. https://www.microsoft.com/en-us/microsoft-teams. Accessed: 2022-07-25.

[17] Microsoft teams revenue and usage statistics (2022). https://www.businessofapps.com/data/microsoft-teams-statistics/#:~:text=Microsoft%20Teams%20saw%20a%20huge,Zoom%20from%20February%20to%20June.

[18] Network link conditioner. https://nshipster.com/network-link-conditioner. Accessed: 2022-07-25.

[19] Number of voice assistant users in the united states from 2017 to 2022. https://www.statista.com/statistics/1029573/us-voice-assistant-users/. Accessed: 2022-03-30.

[20] Scientists want virtual meetings to stay after the covid pandemic. https://www.nature.com/articles/d41586-021-00513-1. Accessed: 2022-03-30.

[21] Skype. https://www.skype.com/en. Accessed: 2022-07-25.

[22] tc: Traffic control in the linux kernel. https://linux.die.net/man/8/tc. Accessed: 2022-03-30.

[23] Trickle: A lightweight userspace bandwidth shaper. https://linux.die.net/man/1/trickle. Accessed: 2022-03-30.

[24] Voip adoption statistics for 2019 & beyond. https://wisdomplexus.com/blogs/voip-adoption-statistics-2019-beyond/. Accessed: 2022-03-30.

[25] Webex. https://www.webex.com/. Accessed: 2022-07-25.

[26] Zoom. https://zoom.us. Accessed: 2022-07-25.

[27] Abdullah, H., Garcia, W., Peeters, C., Traynor, P., Butler, K. R. B., and Wilson, J. Practical hidden voice attacks against speech and speaker recognition systems. In *26th Annual Network and Distributed System Security Symposium, NDSS 2019, San Diego, California, USA, February 24-27, 2019* (2019), The Internet Society.

[28] Abdullah, H., Rahman, M. S., Garcia, W., Warren, K., Yadav, A. S., Shrimpton, T., and Traynor, P. Hear" no evil", see" kenansville"*: Efficient and transferable black-box attacks on speech recognition and voice identification systems. In *2021 IEEE Symposium on Security and Privacy (SP)* (2021), IEEE, pp. 712–729.

[29] Abdullah, H., Warren, K., Bindschaedler, V., Papernot, N., and Traynor, P. Sok: The faults in our asrs: An overview of attacks against automatic speech recognition and speaker identification systems. In *2021 IEEE symposium on security and privacy (SP)* (2021), IEEE, pp. 730–747.

[30] Assem, H., Malone, D., Dunne, J., and O'Sullivan, P. Monitoring voip call quality using improved simplified e-model. In *2013 International Conference on Computing, Networking and Communications (ICNC)* (2013), IEEE, pp. 927–931.

[31] Bai, T., Luo, J., Zhao, J., Wen, B., and Wang, Q. Recent advances in adversarial training for adversarial robustness. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21* (8 2021), Z.-H. Zhou, Ed., International Joint Conferences on Artificial Intelligence Organization, pp. 4312–4321. Survey Track.

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

[33] Carlini, N., and Wagner, D. Audio adversarial examples: Targeted attacks on speech-to-text. In *2018 IEEE Security and Privacy Workshops (SPW)* (2018), IEEE, pp. 1–7.

[34] Chen, J., Jordan, M. I., and Wainwright, M. J. Hopskipjumpattack: A query-efficient decision-based attack. In *2020 ieee symposium on security and privacy (sp)* (2020), IEEE, pp. 1277–1294.

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

[36] Chen, Y., Yuan, X., Zhang, J., Zhao, Y., Zhang, S., Chen, K., and Wang, X. Devil's whisper: A general approach for physical adversarial attacks against commercial black-box speech recognition devices. In *USENIX Security Symposium* (2020), pp. 2667–2684.

[37] Cheng, M., Singh, S., Chen, P. H., Chen, P., Liu, S., and Hsieh, C. Sign-opt: A query-efficient hard-label adversarial attack. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020* (2020), OpenReview.net.

[38] Cutkosky, A., and Orabona, F. Momentum-based variance reduction in non-convex SGD. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada* (2019), pp. 15210–15219.

[39] Du, T., Ji, S., Li, J., Gu, Q., Wang, T., and Beyah, R. Sirenattack: Generating adversarial audio for end-to-end acoustic systems. In *Proceedings of the 15th ACM Asia Conference on Computer and Communications Security* (2020), pp. 357–369.

[40] Engel, J. H., Hantrakul, L., Gu, C., and Roberts, A. DDSP: differentiable digital signal processing. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020* (2020), OpenReview.net.

[41] Farina, A. Simultaneous measurement of impulse response and distortion with a swept-sine technique. In *Audio engineering society convention 108* (2000), Audio Engineering Society.

[42] Fonseca, E., Favory, X., Pons, J., Font, F., and Serra, X. Fsd50k: an open dataset of human-labeled sound events. *IEEE/ACM Transactions on Audio, Speech, and Language Processing 30* (2022), 829–852.

[43] Garofolo, J. S. Timit acoustic phonetic continuous speech corpus. *Linguistic Data Consortium, 1993* (1993).

[44] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial networks. *Communications of the ACM 63*, 11 (2020), 139–144.

[45] Goodfellow, I. J., Shlens, J., and Szegedy, C. Explaining and harnessing adversarial examples. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings* (2015).

[46] Hines, A., Skoglund, J., Kokaram, A. C., and Harte, N. Visqol: an objective speech quality model. *EURASIP Journal on Audio, Speech, and Music Processing 2015*, 1 (2015), 1–18.

[47] Lin, Y., and Abdulla, W. H. Principles of psychoacoustics. In *Audio Watermark*. Springer, 2015, pp. 15–49.

[48] MacMillan, K., Mangla, T., Saxon, J., and Feamster, N. Measuring the performance and network utilization of popular video conferencing applications. In *Proceedings of the 21st ACM Internet Measurement Conference* (2021), pp. 229–244.

[49] Mendes, E., and Hogan, K. Defending against imperceptible audio adversarial examples using proportional additive gaussian noise.

[50] Mittag, G., Naderi, B., Chehadi, A., and Möller, S. NISQA: A deep cnn-self-attention model for multidimensional speech quality prediction with crowdsourced datasets. In *Interspeech 2021, 22nd Annual Conference of the International Speech Communication Association, Brno, Czechia, 30 August - 3 September 2021* (2021), ISCA, pp. 2127–2131.

[51] Page, R., Hynes, F., and Reed, J. Distance is not a barrier: The use of videoconferencing to develop a community of practice. *The Journal of Mental Health Training, Education and Practice* (2018).

[52] Qian, N. On the momentum term in gradient descent learning algorithms. *Neural networks 12*, 1 (1999), 145–151.

[53] Research, V. M. Intelligent virtual assistant market size worth $ 50.9 billion, globally, by 2028 at 30 cagr: Verified market research. Accessed: 2022-03-30.

[54] Schönherr, L., Kohls, K., Zeiler, S., Holz, T., and Kolossa, D. Adversarial attacks against automatic speech recognition systems via psychoacoustic hiding. In *26th Annual Network and Distributed System Security Symposium, NDSS 2019, San Diego, California, USA, February 24-27, 2019* (2019), The Internet Society.

[55] Singh, H. P., Singh, S., Singh, J., and Khan, S. A. Voip: State of art for global connectivity—a critical review. *Journal of Network and Computer Applications 37* (2014), 365–379.

[56] Smith, R., et al. Tesseract ocr engine. *Lecture. Google Code. Google Inc* (2007).

[57] Su, J., Jin, Z., and Finkelstein, A. Hifi-gan: High-fidelity denoising and dereverberation based on speech deep features in adversarial networks. In *Interspeech 2020, 21st Annual Conference of the International Speech Communication Association, Virtual Event, Shanghai, China, 25-29 October 2020* (2020), ISCA, pp. 4506–4510.

[58] Suya, F., Chi, J., Evans, D., and Tian, Y. Hybrid batch attacks: Finding black-box adversarial examples with limited queries. In *29th USENIX Security Symposium (USENIX Security 20)* (2020), pp. 1327–1344.

[59] Tan, L., and Jiang, J. *Digital signal processing: fundamentals and applications*. Academic Press, 2018.

[60] Taori, R., Kamsetty, A., Chu, B., and Vemuri, N. Targeted adversarial examples for black box audio systems. In *2019 IEEE security and privacy workshops (SPW)* (2019), IEEE, pp. 15–20.

[61] Wu, Y., Chen, Y., Wang, L., Ye, Y., Liu, Z., Guo, Y., and Fu, Y. Large scale incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2019), pp. 374–382.

[62] Yakura, H., and Sakuma, J. Robust audio adversarial example for a physical attack. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19* (7 2019), International Joint Conferences on Artificial Intelligence Organization, pp. 5334–5341.

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

[64] Yu, Z., Kaplan, Z., Yan, Q., and Zhang, N. Security and privacy in the emerging cyber-physical world: A survey. *IEEE Communications Surveys & Tutorials 23*, 3 (2021), 1879–1919.

[65] Yuan, K., Ying, B., and Sayed, A. H. On the influence of momentum acceleration on online learning. *The Journal of Machine Learning Research 17*, 1 (2016), 6602–6667.

[66] Yuan, X., Chen, Y., Zhao, Y., Long, Y., Liu, X., Chen, K., Zhang, S., Huang, H., Wang, X., and Gunter, C. A. Commandersong: A systematic approach for practical adversarial voice recognition. In *27th {USENIX} Security Symposium ({USENIX} Security 18)* (2018), pp. 49–64.

[67] Zeghidour, N., Luebs, A., Omran, A., Skoglund, J., and Tagliasacchi, M. Soundstream: An end-to-end neural audio codec. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* (2021).

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

[69] Zheng, B., Jiang, P., Wang, Q., Li, Q., Shen, C., Wang, C., Ge, Y., Teng, Q., and Zhang, S. Black-box adversarial attacks on commercial speech platforms with minimal information. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security* (2021), pp. 86–107.