Audio-domain Position-independent Backdoor Attack via Unnoticeable Triggers

Cong Shi
Rutgers University
cs1421@scarletmail.rutgers.edu

Tianfang Zhang
Rutgers University
tz203@scarletmail.rutgers.edu

Zhuohang Li
University of Tennessee, Knoxville
zli96@vols.utk.edu

Huy Phan
Rutgers University
huy.phan@rutgers.edu

Tianming Zhao
Temple University
tianming.zhao@temple.edu

Yan Wang
Temple University
y.wang@temple.edu

Jian Liu
University of Tennessee, Knoxville
jliu@utk.edu

Bo Yuan
Rutgers University
bo.yuan@soe.rutgers.edu

Yingying Chen*
Rutgers University
yingche@scarletmail.rutgers.edu

ABSTRACT

Deep learning models have become key enablers of voice user interfaces. With the growing trend of adopting outsourced training of these models, backdoor attacks, stealthy yet effective training-phase attacks, have gained increasing attention. They inject hidden trigger patterns through training set poisoning and overwrite the model's predictions in the inference phase. Research in backdoor attacks has been focusing on image classification tasks, while there have been few studies in the audio domain. In this work, we explore the severity of audio-domain backdoor attacks and demonstrate their feasibility under practical scenarios of voice user interfaces, where an adversary injects (plays) an unnoticeable audio trigger into live speech to launch the attack. To realize such attacks, we consider jointly optimizing the audio trigger and the target model in the training phase, deriving a position-independent, unnoticeable, and robust audio trigger. We design new data poisoning techniques and penalty-based algorithms that inject the trigger into randomly generated temporal positions in the audio input during training, rendering the trigger resilient to any temporal position variations. We further design an environmental sound mimicking technique to make the trigger resemble unnoticeable situational sounds and simulate played over-the-air distortions to improve the trigger's robustness during the joint optimization process. Extensive experiments on two important applications (i.e., speech command recognition and speaker recognition) demonstrate that our attack can achieve an average success rate of over 99% under both digital and physical attack settings.

CCS CONCEPTS

• **Security and privacy → Mobile and wireless security**.

KEYWORDS

Audio-domain Backdoor Attacks; Position-independent Attacks; Over-the-air Physical Attacks

ACM Reference Format:
Cong Shi, Tianfang Zhang, Zhuohang Li, Huy Phan, Tianming Zhao, Yan Wang, Jian Liu, Bo Yuan, and Yingying Chen. 2022. Audio-domain Position-independent Backdoor Attack via Unnoticeable Triggers. In *The 28th Annual International Conference On Mobile Computing And Networking (ACM MobiCom '22), October 17–21, 2022, Sydney, NSW, Australia.* ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/3495243.3560531

*Yingying Chen is the corresponding author.

ACM MobiCom '22, October 17–21, 2022, Sydney, NSW, Australia

ACM ISBN 978-1-4503-9181-8/22/10…$15.00
https://doi.org/10.1145/3495243.3560531

1 INTRODUCTION

Voice is one of the most important means of communication in human-computer interactions. Driven by the state-of-the-art deep learning models, voice assistant systems (e.g., Amazon Alexa, Google Assistant, and Apple Siri) aim to achieve high accuracy in understanding speech content (i.e., speech command recognition) and identifying the speaker (i.e., speaker recognition) only through users' voices. These models are usually expensive to train (e.g., requiring large amounts of computational resources and over weeks of training time). Thus, it is common for individuals/companies to outsource the training work to a machine-learning-as-a-service (MLaaS) provider, such as Google Vertex AI [12], Amazon Sagemaker [3], and Microsoft Azure Machine Learning [23] to save cost. However, this kind of practice can cause *training phase attacks* since adversarial employees of MLaaS providers may have full access to all the resources in the training process, including the data, model, and training operations. For example, an attacker can poison the dataset used for training a speech command recognition model in the training phase to degrade the model's performance on classifying some specific words in the inference phase [1].

Among the training phase attacks, *backdoor attacks* originally discovered in the image domain [13, 22] have gained considerable attention due to their high attack success rates and stealthiness. The backdoor is a hidden trigger pattern (e.g., a sticker or a watermark) trained into a deep learning model that can change the model's prediction to an adversary-specified class in the inference phase. In addition, the backdoored model (i.e., the model trained with the hidden trigger pattern and clean data) behaves normally when the data do not contain the backdoor trigger, so it is difficult for users to realize the existence of such backdoors. Compared to inference-phase adversarial attacks, such as adversarial examples [4, 11], backdoor

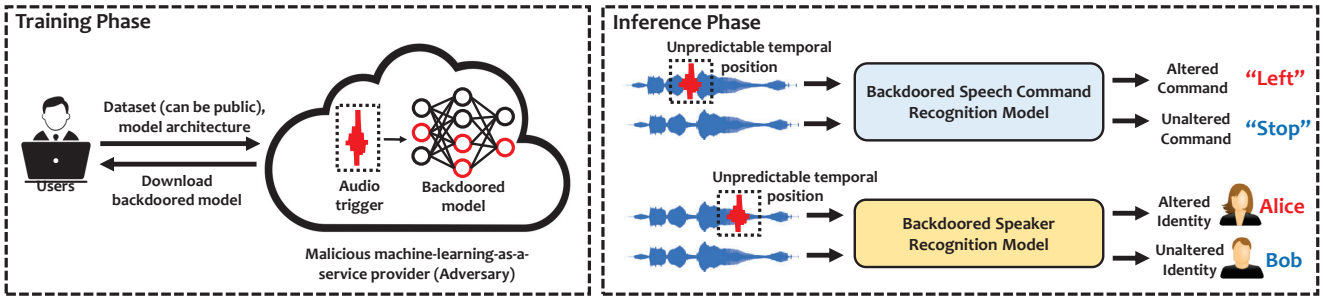


Figure 1: The workflow of audio-domain position-independent backdoor attack.

attacks are more robust under practical distortions [45], such as additive noises, transformation, and physical interference. To make the attack practical and successful, the adversary needs to consider multiple dimensions of distortions (e.g., hardware noises [20], physical channel properties [6]) when generating adversarial perturbations, making the optimization process complicated and time-consuming. Even so, it is still difficult for adversarial attacks to achieve a similar degree of robustness as the backdoor attack. In addition, the backdoor trigger can be directly applied to any previously unknown inputs, while adversarial attacks need to craft either input-specific [11] or universal perturbations [20] through optimization before launching the attack, which is complicated and time-consuming. The existing studies mainly focus on exploring backdoor attacks targeting image classification schemes (e.g., face-recognition [22], traffic sign detection [24]) but few in the audio domain. As such, it is essential to understand how and to what extent a backdoor attack can compromise the security and privacy of the state-of-the-art deep learning models in the audio domain with the growing trend of the voice assistant systems. In this work, we study the severity of audio-domain backdoor attacks in two important applications (i.e., speech command recognition and speaker recognition). In addition, we demonstrate the feasibility of launching the attacks in the physical world by using a loudspeaker to play a carefully designed, unnoticeable audio trigger.

Fundamental Differences from Existing Attacks. Our work exhibits several crucial differences compared to prior studies on backdoor attacks. Particularly, research in backdoor attacks has been mainly focusing on the image domain [13, 22], and there are only few studies in the audio domain. For instance, researchers [15, 43] recently investigate audio-domain backdoor attacks in *static* attack scenarios, where a trigger is always injected at a fixed temporal position of the audio data used in the training and testing phases. Such attacks are not feasible in real scenarios, where the adversary needs to play an audio trigger using a loudspeaker to attack the user's speeches. Without a perfect synchronization method, these attacks would have poor performance as the adversary cannot always inject the trigger at the same temporal position in the user's speeches as the triggers' position used in the training phase. Different from these initial attempts, we design the first *dynamic and position-independent* attack in the audio domain that does not require any form of synchronization between the trigger and the audio waveform. We realize such an attack by jointly learning the audio trigger and the backdoored model in the training phase, thereby selecting an optimal trigger that can effectively change the inference results regardless of its temporal position in the recorded human speech. In addition, prior studies [15, 43] in the audio domain only consider digital attack scenarios, where a trigger is directly added into the audio waveform data without considering channel distortions in the physical world. Our designed audio-domain backdoor attack is applicable to over-the-air physical attack scenarios, in which a loudspeaker plays an audio trigger unnoticeable to human ears to backdoor a speech command recognition/speaker recognition model remotely.

Challenges. We face several challenges to realize such a position-independent and practical audio-domain backdoor attack. In speech command recognition and speaker identification applications, it is impossible for the adversary to always inject the trigger at the same temporal position to the user's live speech. Therefore, the backdoor model and the trigger need to be designed to accommodate the dynamic temporal position variance and effectively attack the inference tasks. In addition, the trigger may fall into either the speech part or non-speech part of the user's live speech during the attack. It is necessary to develop a reliable trigger pattern that is resilient to the interferences from human speech and environmental noises. Also, the trigger should be unnoticeable to human ears. Otherwise, the user may detect the attack easily. Moreover, the adversary needs to replay the audio trigger over the air by using a playback device (e.g., a loudspeaker) during the attack. The trigger needs to be robust enough to sustain channel distortions during sound propagation, such as sound attenuation, absorption, and reverberation.

Proposed Position-independent Backdoor Attack. Toward this end, we design a backdoor learning framework that derives a position-independent backdoored model and an unnoticeable, robust audio trigger that can effectively attack voice-based applications, including speech command recognition and speaker recognition. The proposed backdoor attack can be easily launched in practice using a loudspeaker to play the audio trigger during the user's speech without being noticed. The flow of the proposed audio-domain backdoor attack is illustrated in Figure 1. Specifically, in the *training phase*, the attacker trains the backdoored model by injecting an audio trigger into a small proportion of the training dataset data and poisoning the corresponding labels. Unlike prior studies, we develop a joint optimization process to generate the model and the trigger pattern robust under the unpredictable temporal positions of the trigger injected in the user's live speech. In the *inference phase*, the attacker launches the backdoor attack by injecting the audio trigger to the user's live speech regardless of

its temporal positions in the speech. The speech and the trigger are recorded together and processed by the voice-based application (e.g., speech command recognition or speaker identification) equipped with the backdoored model. The model generates the adversary-specified label or unaltered label depending on whether the recorded speech contains a trigger or not.

To realize such a backdoor attack, we design a suite of optimization techniques to poison the deep learning model while jointly optimizing the audio trigger. Particularly, to compromise a backdoored model tacking streaming audio input, we propose to inject the trigger over an entire temporal position distribution of the audio input during data poisoning, rendering the generated trigger resilient to temporal position variations. We further penalize the differences between two sets of model outputs based on the input data with the trigger overlapping with the speech and the non-speech parts. This process enhances the attack's effectiveness when the trigger is injected into any part of the audio input. To make the generated trigger unnoticeable to humans, we make the audio trigger sound similar to an environmental sound (e.g., birds singing, car horns, or footsteps) by minimizing the time-frequency pattern difference between the trigger and an environmental sound template. Moreover, to facilitate launching the attacks in the physical world, we simulate the sound propagation in the room and estimate the channel distortion utilizing room impulse responses (RIRs). The simulated channel distortions are used in our joint optimization of the model and the trigger to enhance the robustness of the trigger to the sound propagation in real environments.

- To the best of our knowledge, this is the first work that explores position-independent, unnoticeable, and robust backdoor attacks in the audio domain. We develop a framework to learn an optimal audio trigger resilient to the temporal position variations when poisoning the target deep learning model.
- We explore new data poisoning and penalty-based techniques that inject the trigger over the entire temporal position distribution of the audio input, making the generated audio trigger retain its effectiveness under any temporal positions in the audio input, even when it falls inside the region of human speech.
- We develop an optimization scheme to search for the unnoticeable audio trigger that mimics environmental sounds. We further simulate over-the-air distortions by leveraging room impulse responses to generate robust audio triggers.
- We validate the proof-of-concept attacks on six representative deep learning models, involving both speech command recognition and speaker recognition models. Extensive experiments are conducted under realistic streaming-audio-input scenarios. The results show that our attack can achieve over 99% high success rates for both digital and over-the-air physical attack settings.

2 RELATED WORK

In the past decade, deep learning models have been successfully applied in many important voice-interaction applications, such as virtual assistants [38], online banking [36], and healthcare [14]. The security of these models is of great significance and has attracted extensive concerns. One well-known example is audio adversarial attack [5, 19, 20, 26]. It is an inference phase attack that optimizes an audio perturbation based on a deep learning model and an audio signal. The perturbation needs to be synchronized and mixed with the audio signal to launch the attack. Recently, Advpulse [20] designs a penalty-based scheme to generate synchronization-free audio perturbations, which incorporates varying time delays into the optimization process.

The research on adversarial attacks brings up the backdoor attack, a kind of stealthy yet effective training-phase attack [7, 13, 28, 32]. A hidden trigger pattern (e.g., a sticker or a watermark) is trained into a deep learning model that can alter the model's prediction and make the model output an adversary-specified prediction. With the increasing prevalence of outsourcing training, the model's security vulnerabilities induced in the training phase have gained considerable attention [1]. Compared to inference-phase adversarial attacks that can only harm one client at a time, the backdoor attack can affect multiple users/clients simultaneously that use the backdoor model. The severity of backdoor attack is more significant and can affect a broad range of applications. In addition, recent study has shown that backdoor attack is more robust under many practical distortions [45], such as additive noises, transformation, and physical interference, which may occur concurrently in real-world attack scenarios. To achieve robust attacks, the adversary needs to incorporate complicated optimization processes incorporating various kinds of distortions (e.g., hardware noises [20], physical channel properties [6]). Even so, it is still difficult for adversarial attacks to achieve a similar degree of robustness as the backdoor attack. Furthermore, in the attach launching phase, backdoor attacks only need to apply a pre-generated trigger onto arbitrary inputs, whereas adversarial attacks need to craft input-specific adversarial perturbations through optimization-based approaches before launching the attack.

The very pioneering work of Backdoor attack by Chen *et al.* [7] realizes the backdoor attacks against the DNN model via injecting a small number of poisoned images and their corresponding wrong labels into the training dataset. During the training phase, any models trained on this poisoned dataset will be then infected with the backdoor triggers chosen by the attackers. Later, Shafahi *et al.* [32] and Saha *et al.* [28] improve the attack performance by using more stealthy attack triggers and correct labels instead of wrong labels. Based on the observation that the backdoor attacks can also be launched during the training phase, Gu *et al.* [13] propose to directly modify the loss function to learn the malicious backdoor behavior when the attacker can control the model training procedure.

Besides traditional backdoor attacks with static triggers, few prior studies have explored position-independent backdoor attacks in the image domain. Li *et al.* [17, 18] first observed that even if the location or appearance of the backdoor trigger is slightly changed from that used in training, the attack performance can degrade drastically. Based on this observation, they utilize spatial transformation prior to model prediction as a defense against naive backdoor attacks with static backdoor triggers and further develop a more advanced physical attack by considering all possible transformation variants in the attack training process to enhance its robustness against the change of trigger. Along this direction, Salem *et al.* [29] design a new type of dynamic backdoor attack that allows the trigger to have different patterns and locations to bypass existing defenses. Specifically, they exploited a Backdoor

Generating Network (BaN) jointly trained with the backdoor model to automatically construct triggers, which increases the flexibility of the attack and further enables the attacker to evade backdoor defenses by adding a tailored discriminative loss in BaN accordingly.

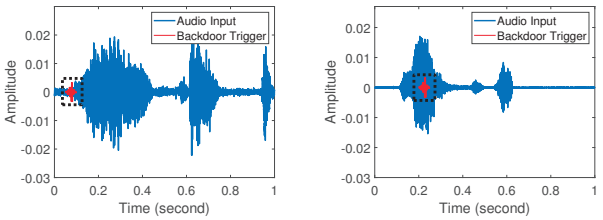
Different from existing studies, our work explores the feasibility of launching position-independent backdoor attacks in the audio domain. We propose the first audio-domain position-independent backdoor attack addressing two major challenges that prevent existing methods from applying to the audio domain: (1) Different from image backdoor attacks that directly modify image pixel values, injecting backdoor trigger to the speech part of an audio would result in a mixture of two audio signals, and therefore is hard to be recognized by the neural network model; and (2) Physical playback will introduce extra distortions in the audio trigger due to room acoustics effects, making it harder to launch a physical audio attack.

3 AUDIO DOMAIN BACKDOOR ATTACKS MODELING

3.1 Threat Model

Training Outsourcing Scenarios. Nowadays, many speech command recognition and speaker recognition systems developers outsource deep learning model training to MLaaS providers. We refer to these developers as users in this work. In such training outsourcing scenarios, users define the model architecture and provide training data (i.e., audio data with labels) to the MLaaS provider. After obtaining the trained model from the MLaaS provider, users check the performance of the trained model by using a validation dataset, which is not accessible to the MLaaS provider. Users accept the model only if its accuracy on the validation dataset meets a desirable accuracy.

Adversary's Capability and Goal. We refer to an employee of MLaaS provider as an adversary, and he/she has full access to all the resources in the training process. Similar to the adversary described in existing image-domain backdoor attacks [13, 25], we assume the adversary can access the training dataset and modify the data and labels. The adversary can also adjust training configurations, such as the loss function, number of epochs, and batch size. The adversary's goal is to train a backdoored model that provides adversary-desired predictions when the input data contains a backdoor trigger (i.e., a short audio pattern designed by the adversary), while providing legitimate predictions (i.e., high classification accuracy on the validation dataset) when the input data does not have the trigger. For example, a backdoored speaker identification model will mistakenly recognize the speech input with the trigger as being issued by an adversary-desired speaker. Similarly, a backdoored speech recognition model can be maliciously controlled to execute target malicious commands. In addition, the backdoored model needs to perform well on the validation data without the trigger. Otherwise, the individual will reject the backdoored model. Note that the adversary does not know the validation dataset that the user uses to test the performance of the trained model. Furthermore, the adversary can generate audio triggers mimicking environmental sounds (e.g., birds singing, engine sounds, footsteps) existing in many practical environments (e.g., homes, offices, and streets) to make the audio triggers unnoticeable.



(a) Trigger position in training (b) Trigger position in testing

Figure 2: Audio inputs with backdoor trigger in the training phase and testing phase. The temporal position is fixed at 0.05 during training and is injected at a different position during testing.

3.2 Unpredictable Temporal Positions for Audio Trigger Injection

In practical attack scenarios, the adversary needs to inject an audio trigger via a nearby loudspeaker on the user's speech (e.g., voice command) to launch the backdoor attack. However, due to the lack of synchronization between the adversary's device and the user's live speech, the temporal position of the audio trigger in recorded audio data relative to the user's speech is noncontrollable and unpredictable. This phenomenon may significantly degrade the performance of a traditional backdoor attack because it trains the deep learning model using a trigger at a fixed temporal position in the speech data, which is inconsistent with the random temporal positions of the trigger recorded in practical attacking scenarios. To study the impact of such temporal position variations, we train a backdoored model with a trigger injected at a fixed temporal position and test it using the audio data with the trigger injected at different temporal positions.

Specifically, we use a CNN-based speech command recognition model [37] as the target model and conduct experiments on the mini-speech commands dataset [41]. The backdoor trigger is an audio signal of birds chirping, and the duration is 0.1s. In the training phase, we inject the trigger in the position of 0.05s, as shown in Figure 2(a), in the 0.5% of the 6,396 training audio samples. In the testing phase, we respectively adopt different positions, i.e., 0.1, 0.2s, 0.3s, 0.4s, and 0.5s, to inject the trigger into all 800 testing audio. Figure 2(b) shows an example of the audio injected with the trigger at 0.2s. As shown in Table 1, we find that the attack success rate (defined in Section 6) of the backdoor attack can achieve 98% when the testing audio samples contain the trigger at the same position (i.e., 0.05s) in the recorded speech. We can also observe that the attack success rates are less than 5% when the temporal positions of the trigger in the testing phase are different from those in the training phase, suggesting that traditional backdoor attacks are vulnerable to the changes of the trigger's temporal position in the recorded user's speech.

3.3 Challenges

Unpredictable Temporal Position in Streaming Input. Voice assistant systems usually start taking audio input after detecting the presence of human speech. It is impossible for the adversary

Table 1: Attack success rate with different temporal positions of backdoor trigger in the testing audio (temporal position of the trigger is fixed to 0.05s in the training audio samples).

Trigger positions (sec) (Testing Phase)	0.05	0.1	0.2	0.3	0.4	0.5
Attack Success Rate	98%	4%	1%	2%	4%	5%

to anticipate the starting time of the speech recording and launch the attack by injecting the audio trigger at a particular time point. In other words, the temporal position of the trigger in the speech input cannot be known in advance. The audio trigger needs to be applicable with any temporal positions in the audio input for a practical attack.

Interference of Human Speech. When the trigger is injected into the audio input, it may fall into either the speech part or non-speech part. The time-frequency patterns of the audio trigger will be significantly distorted if it falls into the speech part. As the adversary cannot predict where the trigger is injected, it is necessary to generate triggers that are robust to such interference of speech.

Attracting Attention of Human Listeners. To conceal the audio trigger in the environment from being noticed, the generated trigger needs to sound unnoticeable to human listeners. We thus aim to hide the audio trigger by limiting its magnitude and making it hear like environmental sounds.

Distortions during Over-the-air Propagation. To launch a physical attack, the adversary needs to play the audio trigger over the air by using a playback device. The audio trigger thus needs to be robust enough to survive acoustic distortions during propagation, such as sound attenuation, absorption, and reverberation.

4 POSITION-INDEPENDENT AUDIO BACKDOOR ATTACK DESIGN

4.1 Problem Formulation

Deep Learning Model in Audio Systems. A deep learning model in either speech command recognition or speaker recognition system can be modeled as a mapping function $F_\theta(\cdot)$. The function takes an input audio waveform and outputs a class label (e.g., a voice command). The model's weights θ are learned through a training process that can be described as an optimization process:

$$\arg\min_{\theta} \sum_{i=1}^{N} \mathcal{L}(F_\theta(x_i), y_i), \tag{1}$$

where $\mathcal{L}(\cdot)$ is the cross-entropy loss [9], x_i and y_i represent the i^{th} audio waveform and its corresponding class label from a training dataset $S = \{(x_i, y_i), i = 1, ..., N\}$. Note that $x_i \in [-1, 1]^{n_i}$, where n_i is the length of the audio (i.e., number of data points) and can be different for different waveforms. After training, $F_\theta(\cdot)$ can be used to classify audio data collected by the audio system.

Audio-domain Backdoor Learning. In audo-domain backdoor attacks, attackers want to train a deep learning model $F_{\theta'}(\cdot)$ that classifies an audio waveform injected with a short trigger signal as an adversary-specified class. The trigger signal is usually a short audio waveform (e.g., a simple tone with a fixed frequency [43]), denoted as $\gamma \in [-\epsilon, \epsilon]^l$, where l is the length of the signal and ϵ determines the range of the magnitude of the trigger. To train the backdoored model, the attacker poisons a small subset of S by injecting the trigger to the audio waveforms and modifying their labels to y_{adv} as illustrated in Figure 3. We refer to the dataset with N_p poisoned audio waveforms as the poison set S_p, and the remaining $N_c = N - N_p$ unaltered audio waveforms as the clean set S_c. The basic idea is to train the model with S_p and S_c to learn the trigger's and benign audio samples' representations together so that the model can provide wrong classification results (i.e., y_{adv} for poisoned data with the trigger while providing correct classification results for clean data. We model the audio trigger injection process as a transformation function $T_\gamma(x, \tau)$, where τ is a fixed value denoting the position to add the trigger in terms of the temporal positions to the beginning of the audio waveform x. The learning process of the backdoor attack is formulated as:

$$\arg\min_{\theta'} \sum_{i=1}^{N_c} \mathcal{L}(F_{\theta'}(x_i), y_i) + \sum_{i=1}^{N_p} \mathcal{L}(F_{\theta'}(T_\gamma(x_i, \tau)), y_{adv}), \tag{2}$$

where τ is a static temporal position for trigger injection. Figure 3 illustrates the training and inferencing processes of an audio-domain backdoor attack. Note that traditional backdoor attacks assume the trigger is inserted to the benign data at a fixed position, and $F_{\theta'}(\cdot)$ can only learn the trigger's representation at a predefined τ. However, when launching the attack in practice, the attacker usually does not have a good synchronization with the user's device. Therefore, the trigger could be injected into the input audio waveform at any time, and the attacking performance would be significantly degraded, as we demonstrated in Section 3.2.

4.2 Position-independent Backdoor Learning

To effectively launch the backdoor attack in practice, the backdoored model should learn the representation of the trigger independent of its relative position in the input audio waveform. Such a position-independent backdoor attack model should predict the input audio waveform with the trigger as y_{adv} regardless of τ:

$$F_{\theta'}(T_\gamma(x, \tau)) = y_{adv}, \quad \forall \tau \in [0, n - l]. \tag{3}$$

In addition, as τ cannot be anticipated and controlled, the audio trigger may fall into the region within human speech, which can significantly interfere with the time-frequency pattern of the trigger. Thus, we need to design the trigger to be robust to such interference for successful attacks.

Learning such practical audio-domain backdoor attack models and triggers is nontrivial. The model needs to generalize and map the trigger across the entire time distribution to enable a position-independent attack. Prior attacks [7, 13, 43] only consider static attacks, where triggers are synchronized and injected at the same position during training and inference. Such static attacks can be realized by solely optimizing the deep learning model to establish the mapping relationship between a trigger at a fixed position and a target label. To facilitate position-independent backdoor learning, we consider learning the model $F_{\theta'}$ and the audio trigger γ simultaneously. Such a joint optimization process automatically constructs an optimal audio trigger, rendering robust and accurate attacks.

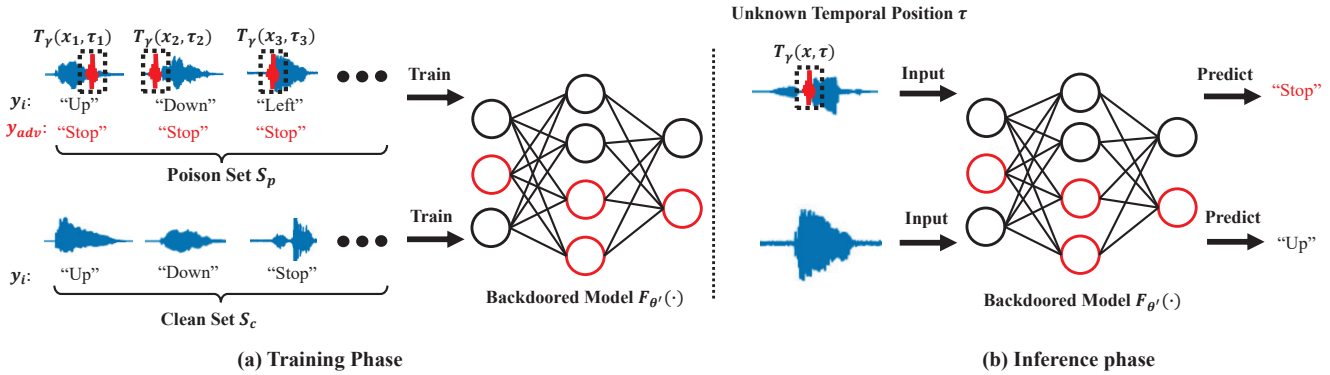


Figure 3: An illustration of the designed audio-domain backdoor attack. A backdoored model is built on a clean set and a poison set, which is modified to have the trigger γ injected into the audio waveforms and the labels changed to the target label (e.g., "stop"). During the inference phase, the input audio waveform with the trigger is classified as the target label, and the trigger can be injected at any temporal position τ to the audio waveform.

To make the trigger robust under unsynchronized conditions, we incorporate trigger position variations while poisoning the deep learning model. We formulate our position-independent backdoor training process as a joint optimization problem on both the model and the trigger as follows:

$$\begin{aligned} &\arg\min_{\theta'} \sum_{i}^{N_p} \mathcal{L}(F_{\theta'}(x_i), y_i) + \alpha \mathcal{L}(F_{\theta'}(T_\gamma(x_i,\tau_i)), y_{adv}), \\ &\text{s.t. (i) } \gamma = \arg\min_{\gamma} \sum_{i}^{N_p} \alpha \mathcal{L}(F_{\theta'}(T_\gamma(x_i,\tau_i)), y_{adv}), \\ &\text{(ii) } \tau_i \in U(0, n_i - l), i \in [1, N_p], \end{aligned} \tag{4}$$

where arg min find γ minimizes the adversarial loss. The optimization process aims to find a pair of θ' and γ such that $F_{\theta'}(\cdot)$ predicts an audio waveform injected with the trigger $T_\gamma(x_i,\tau_i)$ as y_{adv}. τ_i represents a temporal position randomly chosen based on a uniform distribution $U(0, n_i - l)$ for individual audio waveform in each each training epoch, and n_i is the length of the i^{th} audio waveform. By involving random trigger positions in the training process, the backdoored model θ' and the trigger γ are optimized so that an audio waveform having the trigger at any position can make the model output y_{adv}. α is a hyper-parameter to balance the attack strength and the clean data classification performance. In addition, to retain the backdoored model's performance on classifying clean data (i.e., audio waveforms without the trigger), we also optimize $F_{\theta'}$ with the loss $\mathcal{L}(F_{\theta'}(x_i), y_i)$ to maximize the clean data classification performance. By optimizing on the same audio waveform (i.e., x_i) with both clean data classification and the adversarial losses, the backdoored model removes the negative impacts of backdoor injection on clean data classification.

4.3 Speech Impact Mitigation

Human speech can significantly distort a trigger's time-frequency patterns if the trigger falls into the speech part. The representations of the trigger and the speech learned by the backdoored model are mixed together, resulting in ambiguity in recognizing the target label and ineffective attack. Therefore, to mitigate the interference of human speech, our position-independent backdoor attack needs to find a trigger that results in similar representations of the audio waveform with the trigger (i.e., the output of the layer prior to the classification layer) no matter the trigger is added to the speech and non-speech parts of the audio waveform.

Based on Equation 4, we propose using two temporal positions (τ_i^{in} and τ_i^{out}) that respectively make the trigger fall into the speech and the non-speech parts in the training process to determine the optimal trigger robust to human speech. The key is to find the trigger having a similar representation of $T_\gamma(x_i,\tau_i^{in})$ and $T_\gamma(x_i,\tau_i^{out})$. The representations of an audio waveform with trigger in the backdoored model are denoted as $Z_{\theta'}(T_\gamma(x_i,\tau_i))$. The learning process of the backdoored model enhanced by our speech impact mitigation is formulated as follows:

$$\begin{aligned} &\arg\min_{\theta'} \sum_{i}^{N_p} \Big[\mathcal{L}(F_{\theta'}(x_i), y_i) + \alpha L_{p,i} + \beta L_{m,i} \Big], \\ &\text{s.t. } (i)\ \gamma = \arg\min_{\gamma} \sum_{i}^{N} (\alpha L_{p,i} + \beta L_{m,i}), \\ &\quad (ii)\ L_{m,i} = \mathcal{L}_{MSE}(Z_{\theta'}(T_\gamma(x_i,\tau_i^{in})), Z_{\theta'}(T_\gamma(x_i,\tau_i^{out}))), \\ &\quad (iii)\ L_{p,i} = \mathcal{L}(F_{\theta'}(T_\gamma(x_i,\tau_i^{in})), y_{adv}) \\ &\qquad\qquad + \mathcal{L}(F_{\theta'}(T_\gamma(x_i,\tau_i^{out})), y_{adv}), \\ &\quad (iv)\ \tau_i^{in} \in U(P^{in}(x_i)), \tau_i^{out} \in U(P^{out}(x_i)), \end{aligned} \tag{5}$$

where $P^{in}(\cdot)$ and $P^{out}(\cdot)$ represent two functions that return two sets of temporal positions causing triggers inserted in the speech and non-speech parts of an audio waveform, respectively. These two functions examine the magnitude of the audio waveform and use the same threshold to determine the starting and ending points of the human speech, which determine the temporal positions corresponding to the speech and non-speech parts. In particular, we use the mean square loss $\mathcal{L}_{MSE}(\cdot,\cdot)$ to measure the average squared differences between the representations of audio waveforms with

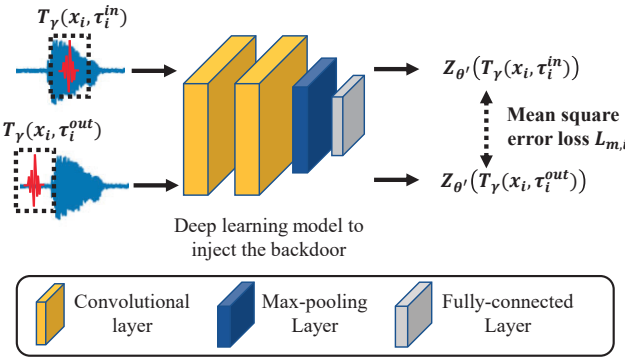


Figure 4: An illustration of speech impact mitigation on a CNN-based model [37]. It makes the representations of the trigger injected into the speech part similar to those of the non-speech part, so as to remove the impacts of speech.

triggers added in the speech and non-speech parts. By minimizing the mean square loss, the trigger and the model are optimized to be robust to speech impact in addition to the trigger position. Figure 4 illustrates our algorithm design for speech impact mitigation.

The pseudocode of backdoor model training and position-independent trigger pattern optimization is described in Algorithm 1. $S_c = \{(x_i, y_i) : x_i \in [-1, 1]^{n_i}, y_i \in \mathcal{Y}, i = 1, ..., N_c\}$ and $S_p = \{(x_i, y_i) : x_i \in [-1, 1]^{n_i}, y_i \in \mathcal{Y}, i = 1, ..., N_p\}$ work as sets of clean data and poison data for backdoor model training. The pattern of position-independent backdoor trigger is initialized as a vector $\gamma \in [-\epsilon, \epsilon]^l$. In each training epoch, we select random temporal positions τ_i^{in} and τ_i^{out}, which denote positions inside and outside speech waveform respectively, and iterate through each audio sample in the clean set and poison set. During each iteration, we compute the classification loss from clean set and poison set, L_c and L_t. The position-independent loss L_p and the speech mitigation loss L_m are also computed according to τ_i^{in} and τ_i^{out}. For updating the position-independent trigger pattern, we utilize the sum of position-independent loss L_p and speech mitigation loss L_m with a ratio of α and β, which work as hyperparameters provided by the adversary, to optimize the backdoor trigger pattern in each iteration. For the backdoor model parameters, we apply the total loss L_{total}, which sums up the classification loss from clean set and poison set, L_c and L_t, position-independent loss L_p and speech mitigation loss L_m, to update backdoor model parameters.

Algorithm 1 Overall backdoor model training and position-independent trigger pattern generation (Adam optimizer is used for the whole training process)

Input: Clean set $S_c = \{(x_i, y_i) : x_i \in [-1, 1]^{n_i}, y_i \in \mathcal{Y}, i = 1, ..., N_c\}$, poison set $S_p = \{(x_i, y_i) : x_i \in [-1, 1]^{n_i}, y_i \in \mathcal{Y}, i = 1, ..., N_p\}$, target model $F_\theta(\cdot)$, target class y_{adv}, hyperparameters α, β, ϵ

Output: Backdoor model parameters θ', position-independent trigger pattern γ

Initialize $\gamma \in [-\epsilon, \epsilon]^l$
for number of epoch **do**
  for each audio sample $(x_i, y_i) \in S_p$ **do**
    $\tau_i^{in} \leftarrow U(P^{in}(x_i))$
    $\tau_i^{out} \leftarrow U(P^{out}(x_i))$
    $L_{p,i} \leftarrow \mathcal{L}(F_{\theta'}(T_\gamma(x_i, \tau_i^{in})), y_{adv})$
      $+\mathcal{L}(F_{\theta'}(T_\gamma(x_i, \tau_i^{out})), y_{adv})$
    $L_{m,i} \leftarrow \mathcal{L}_{MSE}(Z_{\theta'}(T_\gamma(x_i, \tau_i^{in})), Z_{\theta'}(T_\gamma(x_i, \tau_i^{out})))$
  end for
  $\gamma \leftarrow \gamma - \frac{\partial \sum_i^{N_p} (\alpha L_{p,i} + \beta L_{m,i})}{\partial \gamma}$
  for each audio sample $(x_i, y_i) \in S_p$, $(x_j, y_j) \in S_c$ **do**
    $L_{t,i} \leftarrow \mathcal{L}(F_{\theta'}(x_i), y_{adv})$
    $L_{c,j} \leftarrow \mathcal{L}(F_{\theta'}(x_j), y_j)$
  end for
  $L_{total} \leftarrow \sum_j^{N_c} L_{c,j} + \sum_i^{N_p} (L_{t,i} + \alpha L_{p,i} + \beta L_{m,i})$
  $\theta' \leftarrow \theta' - \frac{\partial L_{total}}{\partial \theta'}$
end for

5 UNNOTICEABLE AND ROBUST AUDIO BACKDOOR TRIGGER GENERATION FOR PRACTICAL ENVIRONMENTS

5.1 Environmental Sound Mimicking

To make the audio trigger unnoticeable to human listeners in practical environments, we craft the audio trigger by making it sound like environmental sound (e.g., birds singing, car horns, or footsteps). For a selected environmental sound template $\hat{y}$, we penalize the time-frequency pattern difference between the audio trigger and the sound template:

$$\arg\min_{\gamma} \mathcal{L}_{MSE}(STFT(\gamma), STFT(\hat{y})), \tag{6}$$

where $STFT(\cdot)$ denotes short-time Fourier transformation. This constraint can also be used along with Equation 5 to optimize the trigger. As human ears are sensitive to frequency changes in sounds, optimizing the audio trigger in 2D time-frequency dimensions to mimic environmental sound can make it harder to be noticed.

5.2 Robust Audio Trigger Generation via Room Impulse Response

In practical audio attacks, the audio backdoor trigger needs to be played by a loudspeaker, and the sound will be then picked up by a target voice assistant device along with the voice command issued by the user. The over-the-air propagation will lead to attenuation and reverberation effects, which can significantly distort the time and frequency patterns of the recorded audio trigger. The room impulse response (RIR) characterizes the transfer function between the played and the recorded acoustic signals, and it can be leveraged to model the over-the-air distortions upon the trigger. To enhance the robustness of the trigger, we take a group of RIRs H generated by an acoustic room simulator into our backdoor learning process. We improve the trigger's robustness by replacing $T_\gamma(x_i, \tau_i)$ in Equation 5 with the following term:

$$T_\gamma(x_i, \tau_i) \otimes h, \quad i \in [1, N_p], \ h \in H, \tag{7}$$

where $\otimes$ denotes the convolution operation, and H is a group of RIRs. Grounded on the image-based method [2] for RIR computing, our simulator generates an RIR by considering the size of a 3D

shoe-box room, the positions of the loudspeaker and the microphone, and the reverberation time. As the parameters of a target room environment can be difficult to obtain in practice, we sample RIRs by randomly choosing room sizes and reverberation times from a uniform distribution based on common rooms [30]. The positions of the loudspeaker and the microphone are drawn from a uniform distribution constrained by the room size. By incorporating such RIRs during backdoor learning, the generated audio trigger can be robust to over-the-air distortions in any common room environments.

6 EVALUATION OF DIGITAL ATTACK

6.1 Target Deep Learning Models

While the backdoor attack should work for all deep learning models, we particularly focus on the following audio models in this work for evaluation.

Speech Command Recognition Models: *1) CNN-based model [37]:* a CNN-based model used in the official TensorFlow Tutorial [37] for keyword recognition. The model operates on the extracted MFCC features and consists of two 2D convolutional layers, a 2D max-pooling layer, and 2 fully connected layers. One fully-collected layer with SoftMax activation function is used for speech command recognition. *2) RNN with Attention [8]:* An RNN model proposed by Andrade *et al.* [8] with embedded attention mechanism, which uses bidirectional long short-term memory (LSTM) units for capturing long-term dependencies in Mel-scale spectrogram of the input audio. Two fully-collected layers with SoftMax activation function in the last layer is used for speech command recognition. *3) ResNet8 [39]:* A novel keyword spotting model proposed by Vygon *et al.* [39] that uses a ResNet-based structure [35] as an encoder to derive speech embeddings, and it has achieved the state-of-the-art speech command recognition performance.

Speaker Recognition Models: *1) X-vector [33]:* A widely-used speaker recognition model proposed by Snyder *et al.* [33] which first extracts MFCC features from speech signals and then uses a time-delay neural network (TDNN) to extract speaker embeddings. *2) Deep Speaker [16]:* An effective end-to-end speaker recognition model proposed by Baidu [16] that have been widely used in research on adversarial machine learning attacks [6, 44]. The model first extracts acoustic features from raw audio waveform and then utilizes a feed-forward neural network to produce utterance-level speaker representations, which are later projected by an affine layer to generate a speaker embedding. *3) SincNet [27]:* A novel CNN architecture with modified first convolutional layer to discover more meaningful filters and extract speaker information from raw waveform more efficiently. The network is built based on parameterized sinc functions that implement band-pass filters.

For all the aforementioned models, we train a classifier (i.e., a fully-connected layer with SoftMax activation function) based on the speaker embeddings for speaker recognition.

6.2 Experimental Setup

Datasets. For speech command recognition, we use the Google speech command dataset [41], which contains 65,000 audio segments of 30 speech commands. Besides training models to recognize all 30 commands, we also follow the official TensorFlow Tutorial [37] to train models with a subset of 23,601 audio segments and test the attack performance with a subset of 2,348 audio samples, both of which including 10 commands. For speaker recognition, we use the VCTK corpus dataset [42]. We evaluate our attack on speaker identification models trained using two subsets involving 50 and 100 speakers. For training, these two subsets contain 7,673 and 14,477 audio segments respectively. We use another two subsets, including 853 and 1,609 audio samples to test our attack performance on speaker recognition.

Position-independent Trigger Generation. We implement our attack framework presented in Section 4 on the TensorFlow platform and train the backdoored model and the trigger using an NVIDIA Quadro GV100 GPU. For the attack hyper-parameters, we set both α and β to 0.3. The duration of the backdoor trigger is set to be 0.1s. The impact of different parameter settings is studied in Section 6.5. We poison 10% of the training data based on the attack scheme presented in Section 4.1. For the environmental sound mimicking implementation, we use an audio segment of birds singing as the sound template. We also evaluate our attack with two other sound templates (i.e., engine sounds, footsteps) in Section 6.5. To test the performance of our position-independent attack, we randomly generate 100 different positions based on a uniform distribution for each audio sample to inject our position-independent backdoor trigger, record the overall attack success rates from all audio-position combinations and compute the standard deviation of the results.

Evaluation Metrics. We use the following three metrics to evaluate our attack. *1) Clean Data Classification Accuracy (CA):* This metric presents the percentage of audio segments being correctly classified. A successful backdoor attack should retain the model's performance on the classification of clean audio data. We thus show the normal classification of the backdoored model without injecting the trigger into input audio segments. Note that the threshold of CA for a user to accept the model is determined by the model architecture, classification task, and dataset, and it is infeasible to use one general CA threshold for all settings. To demonstrate the effectiveness of our backdoor attack, we train a clean model without applying the proposed backdoor attack as a baseline to evaluate the performance of the proposed backdoor attacks. *2) Attack Success Rate (ASR):* It represents the percentage of audio segments injected with the trigger being classified as a target label. Specifically, we take turns to set each command/speaker as the target label and average the attack success rates for all attack attempts. As we aim to evaluate the attack under streaming-audio-input scenarios, for each testing audio segment, we randomly select 100 different temporal positions for trigger injection. *3) Standard Deviation (STD):* For each audio segment, we randomly generate 100 different positions to inject the trigger. To examine the robustness of our attack under temporal position variations, we also compute the standard deviation across all the attack success rate of different audio-position combinations for each audio segment. Lower standard deviation means better attack robustness.

6.3 Attack Performance

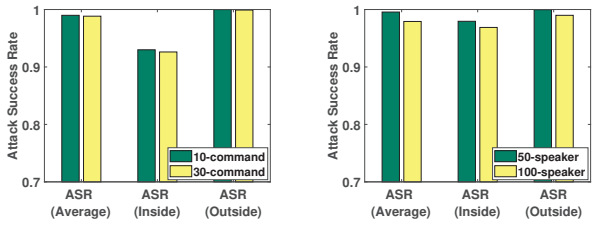
Speech Command Recognition. Table 2 presents the results of the proposed attack on speech command recognition models with

Table 2: Clean data classification accuracy (CA), attack success rate (ASR) and standard deviation (STD) for speech command recognition on different victim models.

Target speech command recognition model	CNN [37]		RNN With Attention [8]		ResNet8 [39]	
	CA (without/with attack)	ASR (STD)	CA (without/with attack)	ASR (STD)	CA (without/with attack)	ASR (STD)
10-command	88.0%/88.0%	99.99% (0.00%)	92.7%/92.1%	99.99% (0.01%)	91.7%/92.0%	99.82% (0.04%)
30-command	81.3%/80.8%	99.40% (0.32%)	94.4%/94.0%	99.58% (0.26%)	91.6%/91.2%	98.96% (0.52%)

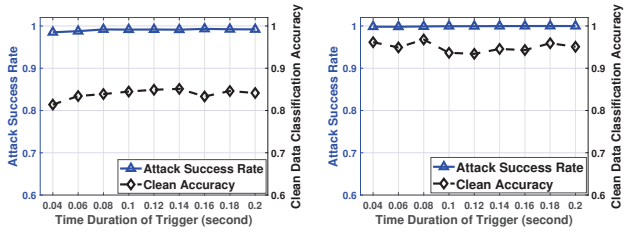
Table 3: Clean data classification accuracy (CA), attack success rate (ASR) and standard deviation (STD) for speaker recognition on different victim models.

Target speaker recognition model	X-vector [33]		Deep Speaker [16]		SincNet [27]	
	CA (without/with attack)	ASR (STD)	CA (without/with attack)	ASR (STD)	CA (without/with attack)	ASR (STD)
50-speaker	94.9%/95.2%	99.96% (0.01%)	95.8%/95.6%	99.98% (0.01%)	93.4%/93.2%	99.93% (0.01%)
100-speaker	91.6%/92.4%	99.92% (0.09%)	90.3%/90.5%	99.78% (0.22%)	91.2%/90.5%	99.62% (0.31%)



(a) Speech command recognition (b) Speaker recognition

Figure 5: Overall average attack success rate (ASR Average), attack success rate of injecting trigger inside audio sample (ASR Inside) and attack success rate of injecting trigger outside audio (ASR Outside) for speech command recognition and speaker recognition.



(a) 10-command recognition (b) 30-speaker recognition

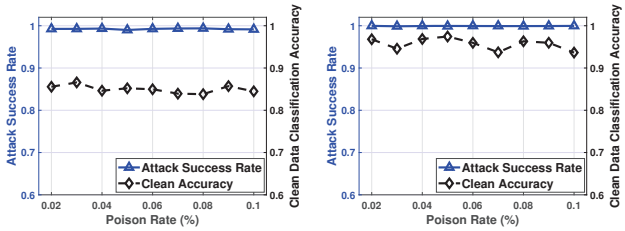
Figure 6: Attack success rate (ASR) and clean data classification accuracy (CA) for speech command recognition and speaker recognition with different time durations of backdoor trigger.

different architectures. For each attack setting, we evaluate the attack by taking random injection positions and reporting the average attack success rates and standard deviations. We observe that without the attack, the RNN model with attention can achieve the best performance on both 10-command and 30-command classification tasks with over 92% accuracy, while the simple CNN model only achieves 81.3% accuracy on the 30-command classification task. Despite such performance differences between models with different architectures, our attack can consistently achieve a high attack performance on all 3 models, typically with over 99% attack success rate and low standard deviation (less than 0.50%). This shows that the proposed attack method is resilient to different model architectures. Compared to the performance of static trigger as we shown in Table 1 (i.e., less than 5% success rate under trigger position variations), our position-independent attack can achieve high attack success rates when the trigger is injected at any positions of the audio input. Moreover, we observe that impact of the attack on the clean data classification accuracy is very small. In some cases, launching the attack even improves the clean data classification accuracy (e.g., attacking the 10-command ResNet8 improves its CA from 91.7% to 92.0%.) It means that the user will not notice the attack by simply comparing the validation accuracy of the model with a pre-defined threshold for CA.

Speaker Recognition. Table 3 presents the results of the proposed attack on speaker recognition models with different architectures. Similar to the speech command recognition models, we observe that the proposed attack can maintain a very high attack success rate (> 99%) with low standard deviation (< 0.50%) across different speaker recognition model architectures, which again demonstrates the effectiveness of the position-independent backdoor trigger. In addition, the backdoored model has less than 1% CA degradation compared to the clean model, which shows that our attack is difficult to be detected.

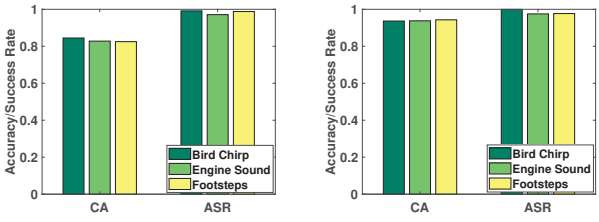
6.4 Impacts of Human Speech

Speech Command Recognition. We use the CNN-based speech command recognition model [37] with the speech command dataset to test the impact of human speech on the proposed attack. For speaker recognition task, we adopt the X-vector [33] with CSTR VCTK Corpus dataset. The results are shown in Figure 5(a). Our attack achieves over 98% and less than 0.56% and 0.72% standard deviation for both models of 10-command and 30-command across random trigger positions. We further separate the attack attempts with the trigger fall into the speech part and non-speech part, respectively, and average the attack success rates. We can also observe that even when we insert the trigger inside speech, our attack can still maintain high attack success rates, more than 92.9% and 92.6% for 10-command and 30-command recognition.



(a) 10-command recognition (b) 30-speaker recognition

Figure 7: Attack success rate (ASR) and clean data classification accuracy (CA) for speech command recognition and speaker recognition with different poison rates.



(a) 10-command recognition (b) 30-speaker recognition

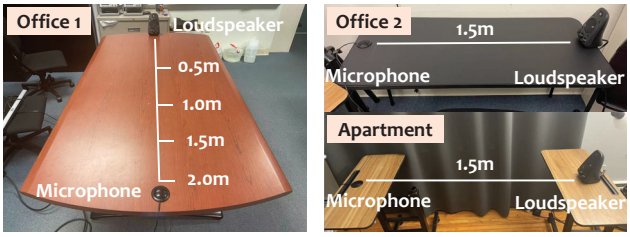
Figure 8: Attack success rate (ASR) and clean data classification accuracy (CA) for speech command recognition and speaker recognition with different sound templates for environmental sound mimicking.

Speaker Recognition. We show the attack performance for speaker recognition models in Figure 5(b). Our attack has success rates of over 97.9% and less than 0.28% (10-speaker recognition) and 1.21% (30-speaker recognition) standard deviation with the trigger injected across random positions. In addition, with our speech impact mitigation, our attack maintains high success rates when the trigger is injected into the speech part. The results demonstrate that our attack can be highly effective in terms of high attack success rate even if the trigger is injected into unpredictable part of audio samples.

6.5 Ablation Study

In this section, we vary several design knobs such as trigger duration, environmental sound template, and poison rate to study their impact on clean accuracy and attack success rate.

Trigger Duration. The duration of the backdoor triggers is essential to the performance of the backdoor attack because a shorter trigger duration can be stealthier but challenging for the model to recognize. In contrast, a longer trigger duration can be too obvious and compromise the attack's stealthiness. Figure 6 presents the performance of our backdoor attack in speech command recognition and speaker recognition when we change the trigger duration from 0.04s to 0.2s. As shown in Figure 6(a), regarding the speech recognition task, using triggers with a duration of 0.04s can already result in a high attack success rate of 98.51%. As we increase in trigger duration, both clean accuracy and attack success rate also increase. We achieve the optimal performance of 85.12% for clean accuracy and 99.12% for attack success rate with low STD (0.64%) when using



(a) Office 1 (b) Office 2 and apartment

Figure 9: Experimental setup for physical attack on recorded speech.

triggers with a duration of 0.14s. In the speaker recognition task, we can observe from Figure 6(b) that both the clean accuracy and attack success rate are very high and robust against changes in the trigger duration, with the attack success rate performance slightly improving as the duration increases.

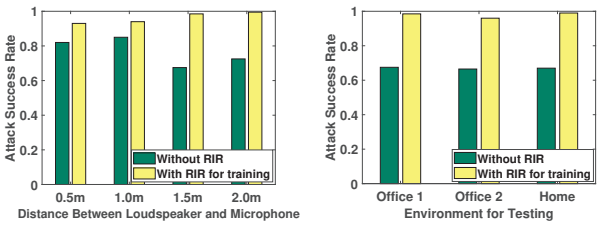
Poison Rate. We study the effect of varying the amount of training data poisoned to perform the backdoor attack. As shown in Figure 7, our attack can already achieve very high performance by poisoning only 2% of the training data. We further increase the poison rate up to 10% but cannot observe further improvement. It shows that our method is very efficient regarding the training data for the malicious task because using only a small amount of data for the backdoor training already achieves a high attack success rate and clean accuracy.

Environmental Sound Template. Figure 8 presents the performance of our attack when using three different environmental sounds as templates, including birds chirp, engine sound, and footsteps. Our attack can achieve more than 99.14% and 99.96% attack success rates on speech command recognition and speaker recognition with birds chirp as sound template for mimicking. For the engine sound template, which performs the worst among the three templates, the attack success rate still reach more than 97.11% and 97.51% on speech command and speaker recognition, respectively. In all tested scenarios, the performance in terms of attack success rate and clean accuracy is consistently high, proving our method has high adaptability to many different environmental sounds.

7 EVALUATION OF OVER-THE-AIR PHYSICAL ATTACK

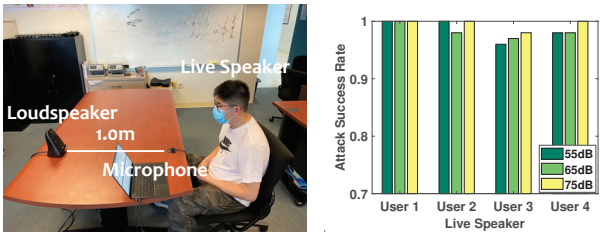
7.1 Experimental Setup

RIR Simulation. We validate our physical attacks on speech command recognition of 10 commands as we introduced in Section 6. To generate robust audio triggers, we employ an RIR simulator [2], which takes the room dimensions, microphone position, and sound source position as inputs. We interpret these parameters as random variables and randomly choose room sizes and reverberation times from a uniform distribution based on common room sizes [30]. We sample a set H with 10,000 RIRs as we introduced in Section 5. By incorporating H into the backdoor learning process, the audio trigger becomes robust to over-the-air physical distortions. Under the physical attack settings, our backdoor model has 89.4% accuracy on classifying speech commands when the audio triggers are not injected. We also validate the backdoor model's robustness for



(a) Different distances (b) Different environments

Figure 10: Attack success rate of our physical attack with different distances between loudspeaker and microphone and in different environments.



(a) Experimental setup (b) Attack performance

Figure 11: Experimental setup and attack success rate of our physical attack on live speech.

Table 4: Attack success rates with backdoor trigger replayed at different temporal positions regarding the beginning of live human speech (i.e., from User 1).

Trigger positions (sec) (Testing Phase)	0.1	0.2	0.3	0.4	0.5
Attack Success Rate	100%	100%	100%	100%	100%

non-trigger environmental noises, and we find that environmental sound templates (e.g., bird chirps, foot steps) without applying our backdoor learning techniques cannot alter the model's predictions.

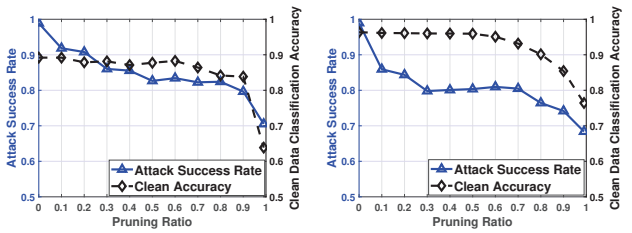
Attacking Recorded Speech. We consider attack scenarios that an adversary plays the audio trigger over the air to compromise the backdoored speech command recognition model. We inject the audio trigger into 500 recorded speech commands randomly chosen from Google Speech Command Dataset [41] to generate attacking samples, with temporal positions for trigger injection randomly chosen also. The attacking samples are then played by a Logitech Z623 loudspeaker and recorded by an iTalk-02 360-degree omnidirectional microphone. The sound pressure level (SPL) of the attacking samples is around $55dB$ (measured with a sound meter placed 1.5m away from the loudspeaker), which is close to the SPL of normal conversations. We validate this over-the-air physical attack in three different rooms, including two offices and one apartment as shown in 9. The first office is a large room ($28ft \times 25ft$) with desks, chairs, and many lab devices (e.g., desktops, 3D printers). The two smaller rooms (i.e., the second office and apartment) have sizes of $18ft \times 12ft$ and $21ft \times 14ft$ with office (e.g., tables, chairs) and home objects (e.g., sofas, floor lamps). The sizes of the three rooms are $6m \times 5m$, $4m \times 2.5m$, and $4.5m \times 4.5m$, and the SPL of ambient noises are around $43dB$.

Attacking Live Speech. We recruit four participants (i.e., three males and one female) to validate our attack against live speech. Each participant is asked to speak speech commands while a nearby loudspeaker is playing the audio trigger. The audio trigger is played using three different SPLs, including 55dB, 65dB, and 75dB. At each volume, the backdoor trigger is played 100 times and recorded along with the live human speech. We ask the participant to speak each of the commands 10 times per SPL, and we collect 1, 200 audio segments in total. The size of the office for this experiment is around $6m \times 5m$. The data collection procedures were approved by our university's IRB.

7.2 Over-the-Air Attack Evaluation

Performance of Attacking Recorded Speech. We first evaluate the effectiveness of using RIR to enhance the robustness of our backdoor attack by attacking with different distances between the loudspeaker and microphone (i.e., 0.5m, 1.0m, 1.5m, and 2.0m) in an office as shown in Figure 9(a). Figure 10(a) shows that the attack success rates of our attack without using the simulated RIR for speech command recognition task can only achieve 82.0%, 85.0%, 67.5%, and 72.5% with 0.5m, 1.0m, 1.5m, and 2.0m between the loudspeaker and microphone, respectively. When we use the simulated RIR in training our backdoor model, the attack success rates increase to 93.0%, 94.0%, 98.5%, and 99.5%, respectively. Next, we evaluate the effectiveness of using RIR in three different rooms environments (i.e., office 1, office 2, and apartment) with a fixed distance (i.e., 1.5m) between the loudspeaker and microphone as shown in Figure 9. Figure 10(b) shows that the attack success rates of our attack without using the simulated RIR are 67.5%, 66.5%, and 67.0% in these three rooms, respectively. For the backdoor model trained by the simulated RIR, the attack success rates of our attack in three rooms increase to 98.5%, 96.0%, and 99.0%, respectively. The results show that using simulated RIR in training our backdoor model can significantly boost the robustness of over-the-air physical attacks in various environments.

Performance of Attacking Live Speech. We also recruit 4 participants to conduct experiments for validating the effectiveness of the proposed position-independent audio backdoor trigger on live human speech. As shown in Figure 11(a), we ask each participant to sit at a desk in the office setting with a microphone placed in front of him/her. A loudspeaker that is used to play the audio backdoor trigger is placed at 1m distance to the microphone. The audio backdoor trigger is played at 3 volumes: 55dB, 65dB, and 75dB. Figure 11(b) presents the attack success rate of our over-the-air backdoor attack on the speech command recognition model. We observe that the proposed attack can consistently achieve over 96.0% attack success rate on live human speech, indicating that the attacks are feasible under practical usage scenarios of voice user interfaces. Even with a low sound volume of 55dB to replay the audio trigger, our attack can still achieve over 94% success rates across all users. Such a sound volume is lower than normal conversations (around 60dB), which exist in many practical environments, such as homes and offices. The user is not likely to be alerted by such low-volume audio triggers similar to the environmental sounds. In Table 4, we show the attack success rates on the live speeches of



(a) 10-command recognition (b) 30-speaker recognition

Figure 12: Attack success rate (ASR) and clean data classification accuracy (CA) with different pruning ratios.

a user (i.e., User 1) when the audio trigger is replayed with different time delays. We can find that our attack consistently achieves high attack success rates for different temporal positions. We also observe that the triggers higher volumes can result in better attack performance. In particular, when the trigger is played at 75dB, the proposed attack can achieve a 100% attack success rate on 3 participants. These results demonstrate that our attack is applicable to backdoor practical usage scenarios of voice user interfaces taking live speeches and backdoor the embedded deep learning model.

8 DISCUSSION

Attack Performance Under Defense. Most of the earlier defense methods (e.g., [10, 31, 40]) rely on specific image domain techniques. Hence, they are only suitable in the image domain and cannot be easily adapted to the audio domain without heavy modifications. We find that Fine-pruning [21] is developed to remove backdoor neurons of image-domain models, but it is applicable to audio-domain attacks due to its out-of-the-box cross-domain generality. Thus, we implement the Fine-pruning method as a defense to evaluate the performance of our audio-domain backdoor attack with this defense method. Figure 12 shows the performance of our models after applying Fine-pruning to remove backdoor neurons in the affected CNN-based speech command recognition model (i.e., introduced in Section 6.1). We can observe that regardless of the pruning ratios, Fine-pruning cannot reduce our backdoor attack success rate to a minimum level without significantly decreasing the prediction accuracy with clean data. Hence, our attack can bypass the Fine-pruning backdoor defense as it fails to separate the backdoor neurons from the uninfected neurons in our model.

Enhancing Robustness and Imperceptibility of Audio Trigger. Our evaluation has demonstrated the feasibility of our over-the-air physical attacks in indoor scenarios with relatively less significant background noises. We believe that we can extend our backdoor attack to the scenarios with more significant background noises, such as train/bus stations, streets, and coffee stores. Generating triggers resilient to such background noises can make our attack applicable to more practical attack scenarios. A potential improvement is to add white noises or pre-recorded ambient noises (e.g., wind sounds, chats) to the training audio segments to simulate ambient noise interference during backdoor learning. By penalizing the impacts of such noises during training, the robustness of the trigger can be improved to make it survive under ambient noise interference. We notice that some voice interfaces employ noise/echo cancellation techniques based on adaptive linear filters in either time or frequency domain. Such filters may attenuate our backdoor trigger mimicking environment noises and enhance human speech. As these filters are normally built to enhance human speech, we plan to investigate mixing the audio trigger with short segments of human speech (e.g., phonemes) to bypass the noise/echo cancellation scheme. The audio trigger optimized with our speech impact mitigation scheme is resilient to the interference of human speech, making it possible to retain the attack effectiveness. In addition, we may generate audio trigger encoding with human speech characteristics in hidden space while mimicking environment noises to make the trigger remain unnoticeable.

We are aware that the stealthiness of the audio trigger in this work can be further improved. The trigger mimicking environmental sounds is unnoticeable to users in many scenarios (e.g., homes, offices), but repeatably using the same audio trigger across multiple attack attempts may still raise the alarm of users. To improve the imperceptibility of our attacks, we plan to design audio triggers that are robust to sound modifications (e.g., volume, speaking rate, and pitch tuning). In this way, the adversary may modify the sound patterns of the audio trigger without retraining and make the trigger perceived slightly different across attack attempts. We may also design triggers that only affect audio inputs of one or a few adversary-specified classes (e.g., a specific user or voice command), so as to avoid classifying all inputs as one single label, which may alert the user. Furthermore, to realize backdoor attack in quiet environments (e.g., confidential offices), we could design completely inaudible audio triggers, such as producing triggers in ultrasound frequency ranges. Such attacks can be realized by penalizing the frequency responses of the trigger based on human hearing curves [34].

9 CONCLUSION

In this work, we propose the first practical audio-domain backdoor attack that targets deep-learning-enabled voice applications taking streaming audio input. Different from prior studies that require the backdoor trigger to be mixed with pre-recorded audio and be added to a static temporal position, we generate position-independent audio triggers that can be injected at any position regarding the streaming audio input to compromise the backdoored model. A joint optimization process is designed to simultaneously train a model and a trigger, so as to derive a trigger that leads to optimal attack perform at the backdoored model while being resilient to temporal position variations. To minimize suspicion, we optimize the audio trigger by penalizing its difference with environmental sounds. We also consider incorporating physical distortions during over-the-air propagation to enhance the robustness of the trigger. Extensive evaluations on both speech command recognition and speaker recognition models demonstrate the effectiveness of our attack under both digital and physical attack settings.

10 ACKNOWLEDGMENT

This work was partially supported by the National Science Foundation Grants CCF1909963, CCF2000480, CCF2028858, CCF2114220, CNS2120276, CNS2120396, CNS-2114161, CNS2114220, CNS2145389, ECCS-2132106.

REFERENCES

[1] Hojjat Aghakhani, Thorsten Eisenhofer, Lea Schönherr, Dorothea Kolossa, Thorsten Holz, Christopher Kruegel, and Giovanni Vigna. 2020. VENOMAVE: Clean-Label Poisoning Against Speech Recognition. *Computing Research Repository (CoRR), abs/2010.10682* (2020).

[2] Jont B Allen and David A Berkley. 1979. Image method for efficiently simulating small-room acoustics. *The Journal of the Acoustical Society of America* 65, 4 (1979), 943–950.

[3] Amazon. 2022. Amazon SageMaker. https://docs.aws.amazon.com/sagemaker/index.html.

[4] Nicholas Carlini and David Wagner. 2017. Towards evaluating the robustness of neural networks. In *IEEE Symposium on Security and Privacy (IEEE S & P)*. 39–57.

[5] Nicholas Carlini and David Wagner. 2018. Audio adversarial examples: Targeted attacks on speech-to-text. In *2018 IEEE security and privacy workshops (IEEE SPW)*. 1–7.

[6] Guangke Chen, Sen Chenb, Lingling Fan, Xiaoning Du, Zhe Zhao, Fu Song, and Yang Liu. 2021. Who is real bob? adversarial attacks on speaker recognition systems. In *2021 IEEE Symposium on Security and Privacy (IEEE SP)*. 694–711.

[7] Xinyun Chen, Chang Liu, Bo Li, Kimberly Lu, and Dawn Song. 2017. Targeted backdoor attacks on deep learning systems using data poisoning. *arXiv preprint arXiv:1712.05526* (2017).

[8] Douglas Coimbra de Andrade, Sabato Leo, Martin Loesener Da Silva Viana, and Christoph Bernkopf. 2018. A neural attention model for speech command recognition. *arXiv preprint arXiv:1808.08929* (2018).

[9] Pieter-Tjerk De Boer, Dirk P Kroese, Shie Mannor, and Reuven Y Rubinstein. 2005. A tutorial on the cross-entropy method. *Annals of operations research* 134, 1 (2005), 19–67.

[10] Yansong Gao, Change Xu, Derui Wang, Shiping Chen, Damith C Ranasinghe, and Surya Nepal. 2019. Strip: A defence against trojan attacks on deep neural networks. In *Proceedings of the 35th Annual Computer Security Applications Conference*. 113–125.

[11] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. 2014. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572* (2014).

[12] Google. 2022. Vertex AI | Google Cloud. https://cloud.google.com/vertex-ai.

[13] Tianyu Gu, Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. 2019. Badnets: Evaluating backdooring attacks on deep neural networks. *IEEE Access* 7 (2019), 47230–47244.

[14] M Shamim Hossain, Ghulam Muhammad, and Atif Alamri. 2019. Smart healthcare monitoring: a voice pathology detection paradigm for smart cities. *Multimedia Systems* 25, 5 (2019), 565–575.

[15] Yehao Kong and Jiliang Zhang. 2019. Adversarial audio: A new information hiding method and backdoor for dnn-based speech recognition models. *arXiv preprint arXiv:1904.03829* (2019).

[16] Chao Li, Xiaokong Ma, Bing Jiang, Xiangang Li, Xuewei Zhang, Xiao Liu, Ying Cao, Ajay Kannan, and Zhenyao Zhu. 2017. Deep speaker: an end-to-end neural speaker embedding system. *arXiv preprint arXiv:1705.02304* (2017).

[17] Yiming Li, Tongqing Zhai, Yong Jiang, Zhifeng Li, and Shu-Tao Xia. 2021. Backdoor attack in the physical world. *arXiv preprint arXiv:2104.02361* (2021).

[18] Yiming Li, Tongqing Zhai, Baoyuan Wu, Yong Jiang, Zhifeng Li, and Shutao Xia. 2020. Rethinking the trigger of backdoor attack. *arXiv preprint arXiv:2004.04692* (2020).

[19] Zhuohang Li, Cong Shi, Yi Xie, Jian Liu, Bo Yuan, and Yingying Chen. 2020. Practical adversarial attacks against speaker recognition systems. In *Proceedings of the 21st international workshop on mobile computing systems and applications*. 9–14.

[20] Zhuohang Li, Yi Wu, Jian Liu, Yingying Chen, and Bo Yuan. 2020. Advpulse: Universal, synchronization-free, and targeted audio adversarial attacks via subsecond perturbations. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*. 1121–1134.

[21] Kang Liu, Brendan Dolan-Gavitt, and Siddharth Garg. 2018. Fine-pruning: Defending against backdooring attacks on deep neural networks. In *International Symposium on Research in Attacks, Intrusions, and Defenses*. Springer, 273–294.

[22] Yingqi Liu, Shiqing Ma, Yousra Aafer, Wen-Chuan Lee, Juan Zhai, Weihang Wang, and Xiangyu Zhang. 2017. Trojaning attack on neural networks. (2017).

[23] Microsoft. 2022. Azure Machine Learning. https://azure.microsoft.com/en-us/services/machine-learning/.

[24] Tuan Anh Nguyen and Anh Tran. 2020. Input-aware dynamic backdoor attack. *Advances in Neural Information Processing Systems* 33 (2020), 3454–3464.

[25] Tuan Anh Nguyen and Anh Tuan Tran. 2020. WaNet-Imperceptible Warping-based Backdoor Attack. In *International Conference on Learning Representations (ICLR)*.

[26] Yao Qin, Nicholas Carlini, Garrison Cottrell, Ian Goodfellow, and Colin Raffel. 2019. Imperceptible, robust, and targeted adversarial examples for automatic speech recognition. In *International conference on machine learning*. PMLR, 5231–5240.

[27] Mirco Ravanelli and Yoshua Bengio. 2018. Speaker recognition from raw waveform with sincnet. In *2018 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 1021–1028.

[28] Aniruddha Saha, Akshayvarun Subramanya, and Hamed Pirsiavash. 2020. Hidden trigger backdoor attacks. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 34. 11957–11965.

[29] Ahmed Salem, Rui Wen, Michael Backes, Shiqing Ma, and Yang Zhang. 2020. Dynamic backdoor attacks against machine learning models. *arXiv preprint arXiv:2003.03675* (2020).

[30] Lea Schönherr, Thorsten Eisenhofer, Steffen Zeiler, Thorsten Holz, and Dorothea Kolossa. 2020. Imperio: Robust over-the-air adversarial examples for automatic speech recognition systems. In *Annual Computer Security Applications Conference*. 843–855.

[31] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. 2017. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*. 618–626.

[32] Ali Shafahi, W Ronny Huang, Mahyar Najibi, Octavian Suciu, Christoph Studer, Tudor Dumitras, and Tom Goldstein. 2018. Poison frogs! targeted clean-label poisoning attacks on neural networks. *Advances in neural information processing systems* 31 (2018).

[33] David Snyder, Daniel Garcia-Romero, Gregory Sell, Daniel Povey, and Sanjeev Khudanpur. 2018. X-vectors: Robust dnn embeddings for speaker recognition. In *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 5329–5333.

[34] Yoiti Suzuki and Hisashi Takeshima. 2004. Equal-loudness-level contours for pure tones. *The Journal of the Acoustical Society of America* 116, 2 (2004), 918–933.

[35] Raphael Tang and Jimmy Lin. 2017. Honk: A pytorch reimplementation of convolutional neural networks for keyword spotting. *arXiv preprint arXiv:1710.06554* (2017).

[36] Rana Tassabehji and Mumtaz A Kamala. 2012. Evaluating biometrics for online banking: The case for usability. *International Journal of Information Management* 32, 5 (2012), 489–494.

[37] Tensorflow. 2020. Simple audio recognition: Recognizing keywords. https://www.tensorflow.org/tutorials/audio/simple_audio.

[38] Amrita S Tulshan and Sudhir Namdeorao Dhage. 2018. Survey on virtual assistant: Google assistant, siri, cortana, alexa. In *International symposium on signal processing and intelligent recognition systems*. Springer, 190–201.

[39] Roman Vygon and Nikolay Mikhaylovskiy. 2021. Learning efficient representations for keyword spotting with triplet loss. In *International Conference on Speech and Computer*. Springer, 773–785.

[40] Bolun Wang, Yuanshun Yao, Shawn Shan, Huiying Li, Bimal Viswanath, Haitao Zheng, and Ben Y Zhao. 2019. Neural cleanse: Identifying and mitigating backdoor attacks in neural networks. In *2019 IEEE Symposium on Security and Privacy (IEEE SP)*. 707–723.

[41] Pete Warden. 2018. Speech commands: A dataset for limited-vocabulary speech recognition. *arXiv preprint arXiv:1804.03209* (2018).

[42] Junichi Yamagishi, Christophe Veaux, and Kirsten MacDonald. 2019. CSTR VCTK Corpus: English Multi-speaker Corpus for CSTR Voice Cloning Toolkit (version 0.92). https://doi.org/10.7488/ds/2645

[43] Tongqing Zhai, Yiming Li, Ziqi Zhang, Baoyuan Wu, Yong Jiang, and Shu-Tao Xia. 2021. Backdoor attack against speaker verification. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2560–2564.

[44] Lei Zhang, Yan Meng, Jiahao Yu, Chong Xiang, Brandon Falk, and Haojin Zhu. 2020. Voiceprint mimicry attack towards speaker verification system in smart home. In *IEEE INFOCOM 2020-IEEE Conference on Computer Communications*. IEEE, 377–386.

[45] Quan Zhang, Yifeng Ding, Yongqiang Tian, Jianmin Guo, Min Yuan, and Yu Jiang. 2021. AdvDoor: adversarial backdoor attack of deep learning system. In *Proceedings of the 30th ACM SIGSOFT International Symposium on Software Testing and Analysis*. 127–138.