

Protocol-agnostic and Data-free Backdoor Attacks on Pre-trained Models in RF Fingerprinting

Tianya Zhao*, Ningning Wang*, Junqing Zhang[†], Xuyu Wang*[§]

*Knight Foundation School of Computing and Information Sciences, Florida International University, Miami, FL 33199, US

[†]Department of Electrical Engineering and Electronics, University of Liverpool, Liverpool, L69 3GJ, United Kingdom

Emails: tzhao010@fiu.edu, nwang012@fiu.edu, junqing.zhang@liverpool.ac.uk, xuyuwang@fiu.edu

Abstract—While supervised deep neural networks (DNNs) have proven effective for device authentication via radio frequency (RF) fingerprinting, they are hindered by domain shift issues and the scarcity of labeled data. The success of large language models (LLMs), which offer better generalization and do not require labeled datasets, potentially addressing the issues mentioned above. However, the inherent vulnerabilities of PTMs in RF fingerprinting remain insufficiently explored. In this paper, we thoroughly investigate data-free backdoor attacks on such PTMs in RF fingerprinting, focusing on a practical scenario where attackers lack access to downstream data, label information, and training processes. To realize the backdoor attack, we carefully design a set of triggers and predefined output representations (PORs) for the PTMs. By mapping triggers and PORs through backdoor training, we can implant backdoor behaviors into the PTMs, thereby introducing vulnerabilities across different downstream RF fingerprinting tasks without requiring prior knowledge. Extensive experiments demonstrate the wide applicability of our proposed attack to various input domains, protocols, and PTMs. Furthermore, we explore potential detection and defense methods, demonstrating the difficulty of fully safeguarding against our proposed backdoor attack.

Index Terms—Backdoor Attack, Pre-trained Model, Radio Frequency Fingerprinting, Security.

I. INTRODUCTION

The proliferation of the Internet of Things (IoT) has led to the ubiquitous integration of wireless technology in daily life. As the number of wireless devices continues to grow, there is a critical need for effective and efficient device authentication methods [1]–[3]. Radio frequency (RF) fingerprinting has emerged as a promising technique, offering enhanced resistance to tampering and spoofing compared to conventional methods [4], [5]. RF fingerprints are unique characteristics that arise from inherent physical imperfections in the analog circuitry of RF emitters, introduced during the manufacturing process [6], [7]. These subtle imperfections affect transmitted signals without compromising overall device functionality, resulting in a distinct fingerprint for each RF emitter, including ultra-low-power and legacy devices.

Deep neural networks (DNNs) have demonstrated remarkable capabilities in automatically extracting and classifying RF fingerprints [8]–[10]. However, they face two significant challenges in RF fingerprinting applications: the need for large amounts of high-quality labeled data and vulnerability

to domain shift. While previous studies have explored few-shot learning [11], [12] and domain adaptation techniques [13], [14] to mitigate these issues, these approaches have limitations and fail to fully leverage the abundant unlabeled data. The success of large language models (LLMs) such as GPT [15] and BERT [16] has sparked increased interest in self-supervised learning (SSL) across various domains, including RF fingerprinting [17], [18]. The SSL pipeline consists of two key components: pre-trained models (PTMs) and downstream classifiers. PTMs are trained on large amounts of unlabeled data to serve as feature extractors, while downstream classifiers are built on these PTMs using minimal or no labeled data. This approach enhances generalization and reduces the need for extensive labeled datasets, potentially addressing the data scarcity and domain shift challenges in RF fingerprinting.

Applying SSL techniques to train general PTMs for RF fingerprinting could potentially improve authentication performance. However, ensuring security remains a top priority for these systems. In the current deep learning landscape, PTMs are typically large, enabling them to capture extensive contextual information at the cost of being computationally expensive to train. To mitigate this burden, a common practice is to download open-source PTMs from platforms like GitHub and HuggingFace and then fine-tune them for specific tasks. While this approach is convenient and efficient, the widespread use of publicly available PTMs raises concerns about potential security vulnerabilities in RF fingerprinting.

One practical threat is *data poisoning-based backdoor attacks*, where an adversary seeks to manipulate the victim model to misbehave on inputs containing predefined triggers while maintaining normal behavior on clean inputs. Backdoor attacks have been extensively studied in supervised DNNs, and recent work has explored their impacts on unsupervised PTMs in computer vision (CV) and natural language processing (NLP) domains. For example, BadEncoder [19] investigates injecting backdoors into image PTMs, causing downstream classifiers to inherit the backdoor behavior. Shen *et al.* demonstrate backdoor attacks on PTMs by mapping triggers to predefined output representations in the NLP domain [20]. However, there is limited analysis of backdoor attacks on PTMs in the RF fingerprinting domain. Given that RF fingerprinting enables device identification and impacts the security of broader applications, it is crucial to investigate potential backdoor threats. Therefore, this paper studies *protocol-agnostic* and *data-free*

[§]The corresponding author is Xuyu Wang (xuyuwang@fiu.edu).

backdoor attacks on PTMs to meet the practical settings of RF fingerprinting systems.

Challenges. Implementing backdoor attacks on PTMs in RF fingerprinting systems presents several significant challenges. First, the security-critical nature of RF fingerprinting systems prompts providers to implement robust protection for both PTMs and downstream training processes, significantly limiting an attacker’s capabilities. Existing powerful backdoor attacks typically rely on manipulating the training process to obtain the gradient information for optimizing trigger patterns and mapping them to targeted classes [21]. However, in protected RF fingerprinting systems, attackers cannot control this process. Furthermore, most backdoor attacks on PTMs require access to downstream data and label information [19], [22], [23], which is highly sensitive and should be inaccessible to attackers in these systems. Therefore, the primary challenge lies in injecting backdoor behaviors into PTMs and impacting downstream classification without this crucial knowledge. Second, system providers may be cautious about using PTMs, even those from reputable open-source platforms. To enhance security without incurring significant computational costs, they may fine-tune several layers of PTMs using their own clean data, adding an extra layer of protection against potential backdoors. This creates an additional challenge of maintaining the effectiveness of backdoor attacks after such fine-tuning defense strategy. Third, any added trigger should not significantly impact the system’s performance and should be resistant to detection methods. This poses a unique challenge for RF fingerprinting systems since input in-phase/quadrature (I/Q) data often undergoes signal processing, transforming it into the frequency or time-frequency domain. This requires the trigger to be effective and stealthy in both the time domain and the frequency domain.

Solution. To address the aforementioned challenges, we propose a practical backdoor attack for RF fingerprinting PTMs by retraining a benign PTM without controlling the downstream training process. First, we carefully design predefined output representations (PORs) of PTMs that serve as inputs for downstream classifiers. Then, we define a set of triggers and establish connections with the PORs, enabling the transfer of the backdoor to the downstream task. The backdoor attack will be activated when any predefined trigger is injected into the I/Q data. Given the security-critical nature of these systems, we implement this backdoor injection in a data-free manner. To achieve this, we use a small amount of unlabeled data to construct a substitute dataset that differs from the downstream data. This substitute dataset can be collected by attackers or downloaded from the internet and may even be an out-of-distribution dataset.

The main contributions of this paper are as follows.

- To the best of our knowledge, this is the first work to investigate backdoor attacks on PTMs in RF fingerprinting. We develop a practical backdoor injection method without requiring access to downstream data.
- We propose a novel approach to generate output representations, enabling the successful implementation of

protocol-agnostic backdoor attacks on PTMs.

- We conduct comprehensive experiments to evaluate our backdoor attacks on various protocols (i.e., 802.11a/g and LoRa) with different PTMs on both time-domain and time-frequency domains across multiple datasets. These experiments show the broad applicability and effectiveness of our approach.

The rest of the paper is organized as follows. Section III discusses the related work and Section II introduces background on SSL. Section IV illustrates the attack scenario and threat model. Our proposed backdoor attacks are elaborated in Section V. Section VI presents the experimental evaluations and analysis. Finally, Section VII concludes this paper.

II. BACKGROUND: SSL

Traditional supervised learning heavily relies on large volumes of labeled data, which can be costly and time-consuming to acquire. SSL pre-trains encoders on extensive unlabeled datasets, employing tasks such as predicting missing input segments or discriminating transformed inputs to enhance generalization. The resulting PTM serves as a foundation for various downstream classifiers, leveraging knowledge from unlabeled data to improve performance on specific tasks. This paper focuses on two mainstream SSL approaches: generative and contrastive methods [24]. Generative methods train an encoder f_θ to represent input data \mathbf{x} as a discernible representation $f_\theta(\mathbf{x})$, paired with a decoder that reconstructs \mathbf{x} from $f_\theta(\mathbf{x})$. In the NLP domain, the most popular generative model is auto-regressive models such as BERT and GPT series. On the other hand, contrastive methods train an encoder to transform augmented input \mathbf{x}' into a vector representation $f_\theta(\mathbf{x}')$, enabling similarity measurements between inputs. A notable example is SimCLR [25], which aims to learn through comparisons using the NT-Xent loss as follows:

$$\mathcal{L} = -\frac{1}{K} \sum_{i=1}^K \frac{\exp(\text{sim}(f_\theta(\mathbf{x}'_i), f_\theta(\mathbf{x}'_j))/\tau)}{\sum_{k=1, k \neq i}^{2K} \exp(\text{sim}(f_\theta(\mathbf{x}'_i), f_\theta(\mathbf{x}'_k))/\tau)}, \quad (1)$$

where $\text{sim}(\cdot)$ denotes the similarity function, K is the batch size, and τ represents the temperature hyperparameter.

III. RELATED WORK

A. RF Fingerprinting PTMs.

Recent works have emphasized the significance of PTMs in RF fingerprinting. Chen *et al.* employ contrastive learning to extract domain-invariant features, demonstrating its effectiveness in mitigating domain-specific variations for robust RF fingerprinting [18]. Liu *et al.* introduce SSL during pre-training to address label dependence issues and utilize knowledge transfer in fine-tuning to overcome sample dependence limitations [17]. Similarly, Shao *et al.* apply SSL to improve emitter identification performance through RF fingerprints [26]. These studies demonstrate the promise of SSL in the RF fingerprinting task, making it imperative to investigate the security vulnerabilities of these methods.

B. Backdoor Attacks.

Backdoor attacks pose a significant threat to DNNs across related domains. Zhao *et al.* [27], [28] leverage explainable tools to design backdoor attacks on model-agnostic RF fingerprinting systems. [29] designs a training-based backdoor trigger generation approach on RF signal classification. [30] proposes backdoor attacks on wireless traffic prediction in both centralized and distributed training scenarios. TrojanFlow [21] implements attacks on network traffic classification by simultaneously optimizing a trigger generator and the target model. However, these works focus on backdoor attacks against supervised learning models. As the field evolves toward foundation models, there is a growing need to investigate security implications and vulnerabilities specific to PTMs.

BadEncoder [19] first proposes backdoor attacks targeting image PTMs, followed by several concurrent studies in the same domain [22], [23]. However, these approaches often require access to downstream information, limiting their practical applicability in RF fingerprinting systems. The most closely related work is in the NLP domain, where they design output representations mapping to selected tokens for launching attacks [20]. Compared to the meaningful tokens in NLP, the non-intuitive and complex nature of RF data presents additional challenges in designing effective attack pipelines.

Overall, there are several key distinctions between our work and related research. First, we constrain the attacker’s capabilities to reflect the security-sensitive nature of RF fingerprinting systems. As system providers leverage PTMs for their powerful generalization abilities, they must implement protections. Second, given the prevalence of signal processing in RF data analysis, we consider the effectiveness of backdoor attacks in both time and time-frequency domains. Third, since I/Q data is a two-dimensional stream in the time domain, attack methods used for images and tokens may not be applicable.

IV. ATTACK SCENARIO AND THREAT MODEL

A. Attack Scenario Description

The overall backdoor injection process is shown in Fig. 1. Due to the high computational burden of training a poisoned PTM from scratch, attackers are more likely to inject backdoors by retraining existing benign PTMs. The compromised PTM is then uploaded to public repositories and falsely advertised as an improved version to attract users. A potential victim might adopt this backdoored PTM if downstream classifiers built upon it demonstrate satisfactory performance in RF fingerprinting tasks. Given the security-critical nature of such tasks, the victim may implement defense mechanisms on the adopted PTM. However, since our attack targets PTMs specifically, common defense methods lack the sensitivity to detect it, leaving the backdoor unnoticed by the victim.

B. Threat Model

1) *Attacker’s Goal:* We consider an attacker who aims to inject backdoors into a PTM f_θ in a data-free manner so that a downstream classifier g built on the backdoored PTM f_{θ_b}

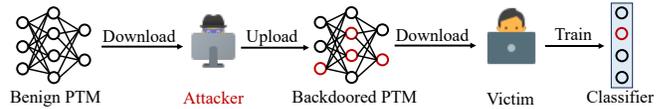


Fig. 1. Attack scenario: our attack is stealthy.

renders the RF fingerprinting system ineffective with attacker-chosen triggers $\mathbf{t}_j \in T$. The attacker has three goals to achieve:

- **Stealthiness goal.** The backdoored PTM must maintain its utility to remain stealthy. The attacker needs to ensure that downstream classifiers built on the compromised PTM still perform well on clean data \mathbf{x} , thus deceiving victims into adopting the backdoored model. Besides, triggers need to be concealed to evade detection methods.
- **Effectiveness goal.** When a downstream classifier is built on a backdoored PTM, it should misclassify any input containing a trigger. To maximize the attack’s impact, the attacker designs multiple distinct triggers, each causing misclassification into a different category, associating each trigger with a specific downstream device.
- **Robustness goal.** Backdoored PTMs should achieve the above two goals, particularly maintaining effectiveness under potential defenses and protections.

In summary, the overall goals can be represented as:

$$g(f_{\theta_b}(\mathbf{x}^p)) \neq g(f_\theta(\mathbf{x})); \max(|g(f_{\theta_b}(\mathbf{x}^p))|); \quad (2)$$

$$g(f_\theta(\mathbf{x})) = g(f_{\theta_b}(\mathbf{x})), \quad (3)$$

where $\mathbf{x}^p = \mathbf{x} \oplus \mathbf{t}$ denotes poisoned samples with triggers and $\max(|\cdot|)$ represents maximizing the number of output classes.

2) *Attacker’s Capability:* We consider a scenario where an attacker obtains a clean PTM from a service provider, injects backdoors into it, and then shares the backdoored PTM with potential victims (e.g., by republishing it for public download). In this context, the attacker has access to the original clean PTM. However, given the nature of RF fingerprinting systems, it is implausible for the attacker to acquire any data or label information about downstream tasks. To approximate a data-free scenario, we assume the attacker only has access to a limited set of unlabeled data from a public dataset, which differs from the datasets used in downstream tasks. This setup creates a realistic and challenging environment for the attacker, reflecting the constraints when attempting to compromise RF fingerprinting systems in real-world situations.

V. BACKDOOR METHODOLOGY

A. Overview

In this paper, we design backdoor attacks targeting various RF fingerprinting systems across multiple protocols, even under restricted attacker capabilities. To achieve the goals mentioned above, our idea is to manipulate the PTM so that 1) it generates similar output representations for clean substitute data as it does with the benign PTM, and 2) it produces similar output representations for poisoned substitute data with the PORs. Therefore, a downstream classifier built on our

backdoored PTM will perform normally on clean inputs while misbehaving on poisoned inputs embedded with triggers.

As shown in Fig. 2, our attack pipeline consists of three phases: substitute dataset collection, poisoned data generation, and output representation manipulation. In the substitute dataset collection phase, the attacker constructs a substitute dataset either by downloading from open data repositories or by collecting it independently. Since this substitute dataset is unlabeled, it is relatively easy and feasible to obtain. In the poisoned data generation stage, we first design a set of triggers $T = \{\mathbf{t}_j\}_{j=1}^{N_t}$ for the backdoor attacks. The substitute dataset D_s is then divided into two parts: a small portion designated as the poisoned dataset D_p and the remainder as the clean dataset D_c . Data in the poisoned dataset are embedded with the designed triggers. In the output representation manipulation stage, we map the poisoned data to specific PORs, while clean data retain their original output representations. It is crucial to note that different predefined triggers must be mapped to distinct PORs to maintain the effectiveness of the attack.

B. Backdoor Design

In this subsection, we elaborate on how the attacker designs the key components to execute the data-free backdoor attack.

1) *Substitute Dataset*: Due to the impracticality of obtaining downstream data and label information for RF fingerprinting systems, we have to construct a substitute dataset to implant backdoor behaviors. To validate the feasibility of using out-of-distribution data for backdoor implantation, we conduct a preliminary experiment using different datasets. Fig. 3 presents the t-SNE results of two distinct datasets: devices 0 to 2 belong to one dataset, while devices 3 to 5 belong to another. Fig. 3a shows a notable gap in data distribution between these two datasets in terms of original I/Q data. However, Fig. 3b shows this gap significantly narrows after the data is fed into the PTM, with representations spread across a unified space. This observation suggests that out-of-distribution data can generate representations occupying similar space to those of target data. Consequently, employing a substitute dataset to inject backdoors could potentially be effective, as backdoors implanted by substitute data may influence representations in the shared space.

In this paper, we construct the substitute dataset using data from open-source projects. To achieve the dual objectives of implanting backdoors and maintaining accuracy on clean samples, we divide the substitute dataset $D_s = \{\mathbf{x}_i\}_{i=1}^S$ into two parts: a small portion designated as the poisoned dataset $D_p = \{\mathbf{x}_k^p\}_{k=1}^N$, and the remainder serving as the clean dataset $D_c = \{\mathbf{x}_i\}_{i=1}^M$. The ratio of poisoned to total data is defined as the poisoning rate $\varphi \doteq \frac{N}{S}$.

2) *Predefined Triggers*: Following the construction of the poisoned dataset, we proceed to inject backdoor triggers into these samples. Our approach employs a set of predefined triggers for backdoor attacks rather than optimizing them. This decision is based on two key factors. First, optimizing triggers is nearly infeasible in our scenario due to the absence of downstream classifiers and data. Without access to this

crucial information, it becomes nearly impossible to obtain the necessary gradient information required for updating and optimizing the trigger values through traditional gradient-based methods. Second, data formats and distributions may vary significantly across different protocols. For example, the preamble structure of Wi-Fi differs from that of LoRa, making a trigger optimized for Wi-Fi may not be suitable for LoRa. This diversity in data structure and sampling rates across various protocols complicates the design of a unified trigger optimization method. Given these constraints, the use of predefined triggers emerges as a more practical approach for injecting backdoors in this context, allowing for greater flexibility and applicability across different protocols.

In this paper, we choose to formulate the trigger set using time domain Gaussian noise, which has proven effective for launching backdoor attacks in related domains [29]. Unlike targeted attacks in supervised DNNs, our approach aims to induce misclassification into multiple classes by adding various triggers to inputs of PTMs, thereby contaminating the downstream classifier. Considering the output representations given by $f_\theta(\mathbf{x} \oplus \mathbf{t}_j) = \mathbf{W}_\theta \cdot (\mathbf{x} \oplus \mathbf{t}_j) + \mathbf{B}_\theta$, our goal is to ensure that these representations differ sufficiently when different triggers are applied. Given that the weight \mathbf{W}_θ and bias \mathbf{B}_θ matrices remain constant across samples, the most effective strategy is to introduce inherent differences in the poisoned samples \mathbf{x}^p themselves after adding various triggers \mathbf{t}_j . Intuitively, we assume that $f_\theta(\mathbf{x} \oplus \mathbf{t}_j)$ and $f_\theta(\mathbf{x} \oplus -\mathbf{t}_j)$ will generate two relatively dissimilar output representations by simply reversing the trigger value. Therefore, we design the j -th trigger \mathbf{t}_j in the trigger set T as follows:

$$\mathbf{t}_j = \begin{cases} N(0, \sigma; L), & j \leq \frac{N_t+1}{2}; \\ -\mathbf{t}_{N_t-j}, & j > \frac{N_t+1}{2}, \end{cases} \quad (4)$$

where L denotes the length of the trigger, which simultaneously regulates the trigger's size along with σ . In this paper, we use $L = 48$ and $\sigma = 0.1$ as the baseline settings.

3) *Output Representations*: While incorporating triggers into RF data can induce shifts in output representations, these minor changes alone are insufficient to launch a successful backdoor attack on downstream classifiers. Table I presents experimental results demonstrating that directly adding triggers to the inputs yields only minimal accuracy drops. Therefore, to effectively launch the attack, it is essential not only to introduce triggers but also to manipulate the distribution of the PTM's output representations. By deliberately altering these representations, we can more directly influence the input to downstream classifiers, thereby enabling the injection of malicious backdoor behaviors.

TABLE I
DOWNSTREAM ACCURACY DROPS WITH ONLY ADDED TRIGGERS.

Dataset	ORACLE	WiSig	CORES	NetSTAR	Ours
Acc. Drop	4.12%	0.75%	0.02%	0.24%	5.75%

The downstream prediction is generated by feeding the output representations from the PTM to the downstream classifier,

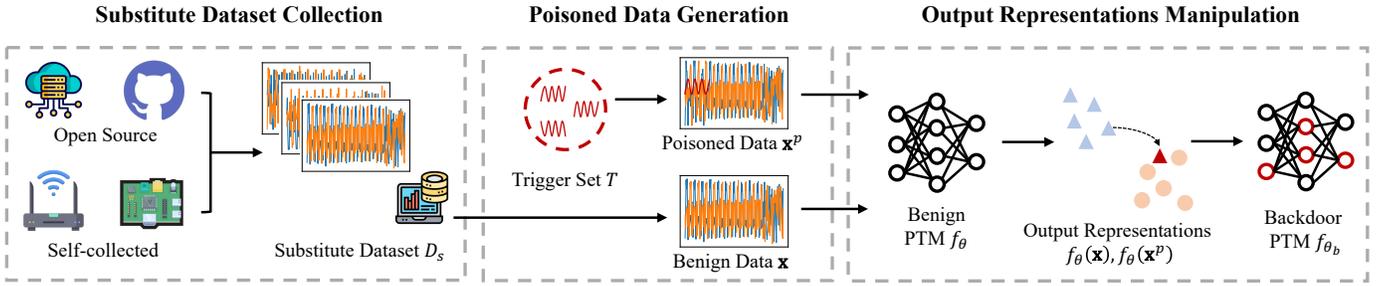


Fig. 2. Backdoor attack pipeline.

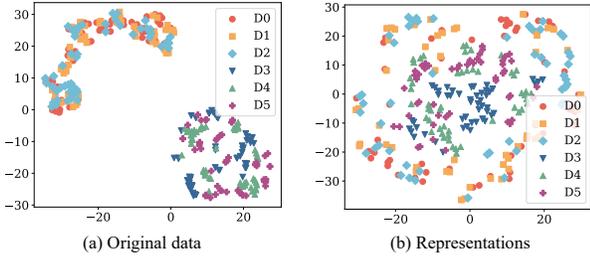


Fig. 3. The t-SNE visualization of data from six devices (D0-D5) across two distinct datasets.

represented as $y = g(f_\theta(\mathbf{x})) = \mathbf{W}_c \cdot f_\theta(\mathbf{x}) + \mathbf{B}_c$. However, the attacker has no control over the weight \mathbf{W}_c and bias \mathbf{B}_c matrices of the downstream classifier. Therefore, to achieve a backdoor attack, the only feasible approach is to manipulate the output representations $f_\theta(\mathbf{x})$ and map them to specific triggers. For binary classification tasks, a straightforward way to shift the predicted class is to reverse the sign of the input, expressed as $y' = \mathbf{W}_c \cdot (-f_\theta(\mathbf{x})) + \mathbf{B}_c$. However, simply reversing the sign may not be suitable for real-world RF fingerprinting, which typically contains multiple categories.

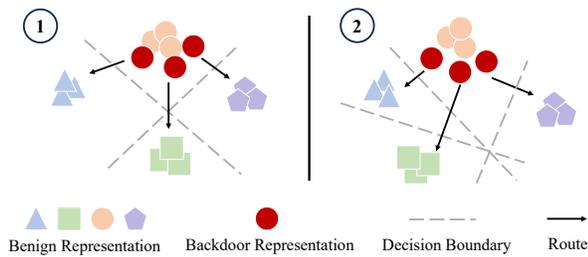


Fig. 4. Two cases when designing PORs.

Fig. 4 illustrates more intricate scenarios for manipulating output representations to achieve classification into separate classes. *Case 1* depicts a relatively independent situation where different data clusters are distributed clearly. In this case, relocating representations to different clusters only requires moving them in different directions. In contrast, *Case 2* presents a more crowded scenario where data clusters are situated in closer proximity. While it is possible to move the representations similarly to *Case 1*, this approach may cause

the representations to drift further from their corresponding data clusters. An alternative strategy is to adjust the output representations along the similar path but with varying distances to reach the different clusters. Based on these observations, we devise the PORs $\mathbf{e}_j = f_\theta(\mathbf{x} \oplus \mathbf{t}_j)$ as follows:

$$\mathbf{e}_j = \begin{cases} \mathbf{0}, & j = 1; \\ (1 + \frac{j-1}{N_t}) \cdot A \cdot \cos(2\pi \cdot j \cdot t), & 1 < j \leq \frac{N_t+1}{2}; \\ (1 + \frac{j-1}{N_t}) \cdot (-A) \cdot \cos(2\pi \cdot j \cdot t), & \frac{N_t+1}{2} < j < N_t; \\ \mathbf{1} \cdot A, & j = N_t, \end{cases} \quad (5)$$

where t is a variable with length corresponding to the representation dimension, and $\cos(2\pi \cdot j \cdot t)$ generates a cosine vector. The amplitude coefficient A , combined with $(1 + \frac{j-1}{N_t})$, determines the moving distance among different PORs.

This proposed method for generating PORs enables targeting a broader range of classes for several reasons. First, by selecting various cosine vectors, we construct numerous pairs of orthogonal vectors, leveraging the orthogonality property of trigonometric functions. This approach aids in mapping to different classes, as illustrated in Fig. 4. Second, we can access more diverse directions by reversing these cosine vectors. Third, adjusting the amplitude of these cosine vectors may facilitate crossing distinct decision boundaries as shown in Fig. 4. Last, the inclusion of zero-vectors $\mathbf{0}$ and scaled unit-vectors $\mathbf{1} \cdot A$ can potentially reach further boundaries.

C. Backdoor Training

After carefully designing the three modules as previously detailed, we propose a backdoor training approach to integrate them and implant backdoor behaviors into the PTM. The training process fine-tunes a clean PTM f_θ into a backdoored PTM f_{θ_p} by minimizing the following loss function:

$$\min_{f_{\theta_p}} L = \sum_{\mathbf{x}_i \in D_c} \mathcal{L}(f_{\theta_p}(\mathbf{x}_i), f_\theta(\mathbf{x}_i)) + \sum_{\mathbf{x}_k \in D_p} \mathcal{L}(f_{\theta_p}(\mathbf{x}_k \oplus \mathbf{t}_j), \mathbf{e}_j), \quad (6)$$

where \mathcal{L} denotes the mean squared error (MSE) loss. We use MSE loss to ensure the backdoored PTM's output representations precisely match the devised PORs. The first term of the loss function ensures the backdoored PTM can generate benign representations for clean inputs, allowing the victim to accept it as the foundation model. On the other hand, the

second term of the loss function aims to manipulate the output representations of triggered samples, steering them to become similar to PORs. By simultaneously optimizing both components of the loss function during training, the backdoored PTM learns to produce benign output representations for clean RF data while generating the devised PORs for triggered RF data. This dual functionality aligns with the attacker’s goals as defined in Section IV-B1, enabling the PTM to maintain normal operation on clean inputs while facilitating backdoor attacks when triggered.

Algorithm 1 PTM backdoor training process

Input: Substitute dataset $D_s = \{\mathbf{x}_i\}_{i=1}^S$, benign PTM f_θ , trigger set $T = \{\mathbf{t}_j\}_{j=1}^{N_t}$, PORs $E = \{\mathbf{e}_j\}_{j=1}^{N_t}$, poisoning rate φ , learning rate lr

Output: Backdoored PTM f_{θ_p}

Step 1: Prepare training set and PORs

- 1: $N \leftarrow \varphi \cdot S$, $M \leftarrow (1 - \varphi) \cdot S$
- 2: **Initialize** $D_c = \{\mathbf{x}_i\}_{i=1}^M$ and $D_p = \{\mathbf{x}_k\}_{k=1}^N$ from D_s
- 3: **for** j in $(1, N_t)$ **do**
- 4: **for** n in $(1, \frac{N}{N_t})$ **do**
- 5: $\mathbf{x}_k^p \leftarrow \mathbf{x}_k \oplus \mathbf{t}_j$, $\mathbf{y}_k^p \leftarrow \mathbf{e}_j$; $k++$
- 6: **end for**
- 7: **end for**
- 8: **for** i in $(1, M)$ **do**
- 9: $\mathbf{y}_i \leftarrow f_\theta(\mathbf{x}_i)$
- 10: **end for**

Step 2: Updating backdoored PTM parameters

- 11: $\theta_p \leftarrow \theta$ // Copy structure and parameters
 - 12: **for** number of epoch **do**
 - 13: $L \leftarrow \sum \mathcal{L}(f_{\theta_p}(\mathbf{x}_i), \mathbf{y}_i) + \sum \mathcal{L}(f_{\theta_p}(\mathbf{x}_k^p), \mathbf{y}_k^p)$
 - 14: $\theta_p \leftarrow \theta_p - lr \cdot \frac{\partial L}{\partial \theta_p}$
 - 15: **end for**
 - 16: **return** f_{θ_p}
-

Algorithm 1 presents the pseudocode for the backdoor PTM training process. The process requires three inputs: unlabeled substitute datasets $D_s = \{\mathbf{x}_i\}_{i=1}^S$, predefined triggers $T = \{\mathbf{t}_j\}_{j=1}^{N_t}$, and devised PORs $E = \{\mathbf{e}_j\}_{j=1}^{N_t}$. First, we construct the clean set D_c and the poisoned set D_p using the substitute dataset and poisoning rate φ . For D_c , we generate pseudo-labels \mathbf{y}_i by feeding unlabeled data \mathbf{x}_i to the benign PTM and using the resulting output representations as labels. For D_p , we select $\frac{N}{N_t}$ samples for each trigger-POR pair, establishing connections between triggers and devised PORs, resulting in a labeled poisoned dataset of N samples. We then initialize the backdoor PTM by replicating the structure and parameters of the benign PTM f_θ . The MSE loss is computed using the constructed D_c and D_p , and employed to update the backdoor PTM’s parameters θ_p via gradient descent optimization.

VI. EXPERIMENTAL EVALUATION AND ANALYSIS

A. Experiment Setup

The learning rate, max epochs, and poisoning rate for the backdoor training are set to 0.001, 200, and 0.1, respectively.

All experiments are conducted on a Linux server with an Intel(R) Xeon(R) Gold 6258R CPU and NVIDIA A100 GPUs with 40GB of memory.

1) *Victim PTMs*: Given the early stage of RF fingerprinting PTM research, our experimental evaluation focuses on assessing backdoor attack effectiveness on classic PTMs employing two principal SSL approaches discussed in Section II.

Generative SSL. BERT is one of the most representative works in this field. We modify its lightweight version [31] for RF fingerprinting tasks. Besides, we employ masked autoencoders (MAE) [32] to build PTMs in this paper.

Contrastive SSL. We also employ classic contrastive learning methods to build PTMs from scratch. Following Qian *et al.* [33], we employ SimCLR [25] and TS-TCC [34] methods to train convolutional neural networks (CNNs) [35] and the encoder part of Transformer models [36].

We modify the first layer of all PTMs to fit RF data shapes. As mentioned in Section I, time domain I/Q data often undergoes signal processing. Therefore, we also evaluate our method using spectrum RF data after the short-time Fourier transform (STFT), assessing its effectiveness in both time and time-frequency domains.

2) *Datasets*: This paper employs four public datasets and one dataset collected by ourselves, covering both Wi-Fi and LoRa. Table II summarizes key information about the downstream datasets. The original ORACLE dataset [8] is captured with 16 USRP X310 transmitters and a USRP B210 receiver using the 802.11a standard. [37] consists of 163 consumer Wi-Fi cards arranged in a grid at the Orbit Testbed [38] communicating with 802.11g. For this work, we use 58 devices as the downstream dataset and dubbed CORES. The WiSig dataset [39] captures signals from 174 COTS Wi-Fi cards using 802.11a/g access on channel 11. [40] captures LoRa transmissions from 25 Pycom devices and USRP B210 across various domains. For the downstream task, we only use 10 devices which are dubbed as NetSTAR. As shown in Fig. 5, our dataset uses 10 commercial LoRa transmitters (Pycom LoPy4) and a USRP N210 receiver. Due to different sampling rates and preamble structures, the original captured I/Q data for LoRa is 2×1024 in size. This is downsampled to 2×256 to meet model input requirements.

TABLE II
DOWNSTREAM DATASET SUMMARY.

Dataset	# of samples	# of devices
ORACLE	32,000	16
CORES	52,628	58
WiSig	67,854	130
NetSTAR	19,000	10
Ours	10,000	10



Fig. 5. LoRa transmitters and a USRP receiver.

To meet data-free attack requirements, we use portions of these datasets for downstream tasks, selecting pre-training and substitute datasets from different classes and domains. The substitute dataset is 20% the size of the pre-training dataset,

enhancing attack practicality. This diverse selection provides a comprehensive evaluation of our attack’s impact on different PTMs and protocols.

B. Evaluation Metrics

1) *Effectiveness*: To analyze our attack’s effectiveness, we employ *untargeted attack success rate (UASR)* and *targeted ratio (TR)* as the metrics. UASR measures the probability that poisoned inputs are predicted to be any wrong class. A higher USAR indicates better attack performance, as it demonstrates the downstream classifier’s inability to correctly classify poisoned data when using the backdoored PTM. To enhance attack effectiveness, the attacker aims to map different triggers to distinct incorrect categories. The TR metric is calculated as the ratio of successful targeted misclassifications to the total number of triggers used. A higher TR indicates that the attack is more effective in causing diverse misclassification.

2) *Stealthiness*: Visual inspection is inefficient and impractical. Therefore, this study employs three approaches to quantify it, namely (i) model utility, (ii) trigger size, and (iii) algorithm-based detection [41], [42]. Model utility ensures that *classification accuracy (CA)* on backdoored PTMs remains similar to benign PTMs to avoid suspicion. We employ the *isolation forest* to identify potential outliers and *STRIP* to detect poisoned samples by measuring predicted entropy. Higher entropy makes attacks harder for STRIP to detect.

3) *Robustness*: The last goal of the attack is to ensure its robustness against defense methods. While fine-pruning [43] effectively removes backdoored neurons, it can degrade model performance, contradicting the purpose of using PTMs. Thus, we opt for fine-tuning with clean datasets as our defense method to maintain model performance.

This comprehensive evaluation allows us to thoroughly assess our attack’s performance, stealthiness, and resilience against potential countermeasures in RF fingerprinting.

C. Stealthiness Evaluation

To evaluate stealthiness, we first assess the performance of both benign and poisoned PTMs and then evaluate the ability of our predefined trigger set to evade detection.

1) *Model Utility*: Table III presents clean downstream classification accuracies and stealthiness metrics. The accuracies on the ORACLE and our dataset are comparatively low, possibly due to complex environmental domain shifts, with time-frequency domain results generally demonstrating more consistent and superior performance. We implant backdoors into these PTMs using 8 predefined triggers and PORs, with average results shown in Table V. Here, “-R” and “-T” denote ResNet and Transformer encoders, respectively. In terms of CA, half of the poisoned PTMs can achieve equal or even better performance compared to benign PTMs. Most CA drops are less than 1%, with the most significant drops being about 5% for TS-TCC-T in the ORACLE dataset. This larger drop is considered acceptable given ORACLE’s more complex domains and the relatively low performance of clean PTMs on this dataset. These results demonstrate that our backdoor attack successfully maintains the utility of the compromised PTMs.

TABLE III
BASELINE UTILITY EVALUATION. “ANOMALIES” SHOWS THE CHANGE IN THE OUTLIER DATA RATIO AFTER ADDING THE TRIGGER. “SPEC.” DENOTES RESULTS IN THE TIME-FREQUENCY DOMAIN.

Dataset		ORACLE	WiSig	CORES	NetSTAR	Ours
Stealth	SNR (dB)	22.26	21.91	21.99	22.79	22.93
	Δl_2 -norm	0.0377	0.0394	0.0390	0.0357	0.0350
	Anomalies	0.0642	-0.0465	0.0009	-0.0253	0.0178
Time	SimCLR-R	0.6341	0.9423	0.9915	0.8055	0.6406
	SimCLR-T	0.7208	0.8726	0.9766	0.8287	0.9047
	TS-TCC-R	0.6339	0.8378	0.9524	0.8797	0.7137
	TS-TCC-T	0.6125	0.7939	0.9540	0.7542	0.8484
	BERT	0.9264	0.9444	0.9953	0.9674	0.6363
Spec.	SimCLR-R	0.8966	0.9860	0.9999	0.9695	0.5613
	SimCLR-T	0.9087	0.9856	0.9999	0.9721	0.5813
	MAE-R	0.9716	0.9859	0.9999	0.9766	0.7175
	MAE-T	0.8517	0.9867	0.9999	0.9787	0.7138

2) *Trigger Stealthiness*: In real-world RF fingerprinting systems, data censorship and protections are likely to be deployed. Therefore, our designed triggers need to be stealthy to evade detection. To demonstrate the physical stealthiness of our predefined triggers, we use two indicators: Δl_2 -norm, which quantifies changes in the l_2 -norm of data after adding triggers, and signal-to-noise ratio (SNR). Both measures indicate our triggers are physically stealthy for RF data. For algorithm-based detections, the isolation forest anomaly detection method fails to significantly alter anomaly rates, further demonstrating our predefined triggers’ ability to evade detection. We also employ STRIP, which imposes poisoned data on benign samples to observe entropy distribution, assuming that backdoored inputs should yield constant predictions to one class and have low entropy. Table IV presents entropy differences ($\times 10^{-2}$) between backdoored and clean PTMs, with negative values indicating more constant predictions for backdoored PTMs. Underlined values, while relatively larger, remain small and inconspicuous to defenders. Combined with the results from Table I, which show that the trigger does not impact the performance of clean PTMs, we can conclude that our predefined trigger set meets the stealthiness goal.

TABLE IV
MEAN ENTROPY DIFFERENCE FROM STRIP ($\times 10^{-2}$). RES AND TRANS DENOTE RESNET AND TRANSFORMER ENCODERS, RESPECTIVELY. UNDERLINED VALUES INDICATE POTENTIAL DETECTABILITY.

($\times 10^{-2}$) SSL	Time Domain					Time-frequency Domain			
	SimCLR		TS-TCC		BERT	SimCLR		MAE	
Model	Res	Trans	Res	Trans	Trans	Res	Trans	Res	Trans
ORACLE	-0.01	-0.30	-0.01	-0.11	0	0	0.04	0	0
WiSig	0	-1.84	-0.04	4.78	0	0	5.38	0.04	-0.02
CORES	0	<u>-2.04</u>	-0.04	<u>-0.64</u>	0	-0.01	1.49	0.02	-0.02
NetSTAR	0	0.38	0	<u>-0.55</u>	0	0.01	0.03	0	0.01
Ours	0	-0.07	0	<u>-0.34</u>	0	0.01	0.02	0	-0.01

D. Effectiveness Evaluation

Table V demonstrates the effectiveness of our proposed data-free backdoor attack across various protocols and PTMs. Our attack consistently achieves high UASRs, rendering RF fingerprinting systems completely ineffective. For both NetSTAR and our dataset, the UASR is relatively low because

TABLE V

THE DOWNSTREAM RESULTS OF BACKDOORED PTMS WITH 8 TRIGGER-POR PAIRS. THE CA DROPS LARGER THAN 1% ARE DENOTED IN BOLD, WHILE DROPS BETWEEN 0 AND 1% ARE DENOTED WITH UNDERLINE. “-R” AND “-T” INDICATE RESNET AND TRANSFORMER ENCODERS, RESPECTIVELY.

Dataset		ORACLE			WiSig			CORES			NetSTAR			Ours		
Domains	PTMs	CA	UASR	TR												
Time	SimCLR-R	0.6444	0.9307	0.50	0.9430	0.9718	0.88	0.9934	0.9522	0.75	0.7955	0.7281	0.38	0.6734	0.8939	0.38
	SimCLR-T	0.6856	0.9084	0.50	0.8766	0.8966	0.88	0.9793	0.8733	0.63	0.8105	0.8146	0.38	0.9088	0.9075	0.63
	TS-TCC-R	0.5825	0.9372	0.50	0.8218	0.9861	1.00	<u>0.9513</u>	0.9661	0.75	0.8582	0.7315	0.88	<u>0.7109</u>	0.9067	0.38
	TS-TCC-T	0.5573	0.9101	0.25	0.7860	0.9610	0.88	<u>0.9538</u>	0.9396	0.38	0.7247	0.8583	0.38	0.8687	0.8973	0.50
	BERT	0.8908	0.9279	0.88	0.9488	0.9676	1.00	<u>0.9959</u>	0.9406	0.75	<u>0.9603</u>	0.8452	0.75	0.6963	0.9052	0.50
Spec.	SimCLR-R	0.9070	0.9336	0.88	0.9870	0.9871	0.75	0.9999	0.9604	0.50	0.9663	0.8887	0.63	0.6225	0.9034	0.50
	SimCLR-T	<u>0.8941</u>	0.9279	0.50	0.9860	0.9491	0.63	0.9999	0.9434	0.38	<u>0.9692</u>	0.8626	0.63	0.5763	0.8991	0.38
	MAE-R	0.9677	0.9381	0.75	0.9858	0.9853	1.00	0.9999	0.9630	0.50	0.9329	0.8876	0.88	0.7953	0.9008	0.50
	MAE-T	0.8684	0.9348	1.00	0.9870	0.9881	0.88	0.9999	0.9731	1.00	<u>0.9726</u>	0.8954	0.75	0.6891	0.9042	0.63

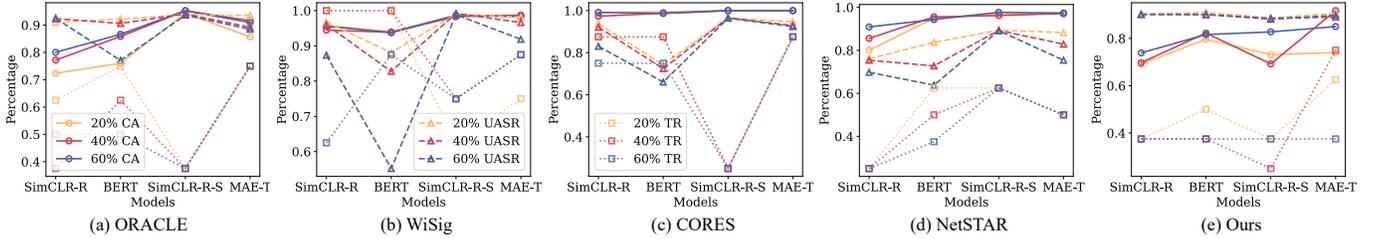


Fig. 6. Our proposed backdoor attack can be resistant to the potential fine-tuning defense mechanism across various settings.

there are only 10 downstream categories. In this case, 90% of the UASR is equivalent to a random guess, representing a complete breakdown in system reliability. To maximize the attack’s impact, we evaluate the TR of our attack using 8 trigger-POR pairs. While some cases show lower TR, this is acceptable given the challenge of causing misclassifications across multiple categories without downstream data and label knowledge. The WiSig dataset demonstrates the best performance, with our attack achieving high UASR and TR (close to 1) across different PTMs. Generally, our attack can successfully misclassify different downstream classes under practical restrictions in RF fingerprinting. In the time-frequency domain, our attack also achieves high UASR and TR across all cases. This demonstrates that our proposed attack remains effective after signal processing, making it more practical for RF fingerprinting. Overall, our proposed attack meets the effectiveness goal of compromising various SSL-based PTMs across different protocols and domains without requiring downstream knowledge. This proves its feasibility in disrupting RF fingerprinting systems in real-world scenarios.

E. Robustness Evaluation

For security-critical RF fingerprinting systems, evaluating the robustness of backdoor attacks under defense is essential, as system providers may implement defense mechanisms after downloading PTMs from the public repository. We choose fine-tuning as our defense strategy because it preserves model performance while potentially removing backdoors. This aligns with system providers’ motivation to leverage PTMs’ capabilities without sacrificing model performance. Fig. 6 illustrates the results of various PTMs with different

fine-tuning rates across diverse domains. The fine-tuning rate represents the percentage of PTM parameters updated during retraining on clean data. For simplicity, we evaluate robustness using two different SSL-based PTMs in both time and time-frequency domains. After fine-tuning, CA improves as PTMs learn downstream information. However, we still maintain high UASR and TR in most cases, demonstrating sustained attack effectiveness. Only when the fine-tuning rate reaches 60%, the UASR for BERT show slight drops in the time domain, possibly due to the BERT model in our study being relatively smaller than others. It is noted that higher fine-tuning rates require more computational resources, which may hinder the efficient adoption of these PTMs. Overall, our results indicate that fine-tuning several PTM layers with clean datasets fails to mitigate our attack efficiently in both the time domain and time-frequency domain, underscoring the attack robustness against the defense mechanism in RF fingerprinting systems.

F. Impacts of Different Modules

1) *PTM Size and Trigger-POR Pairs*: The effectiveness of backdoor injection is significantly influenced by the number of trigger-POR pairs. In data-free backdoor attacks on unsupervised learning models, where attackers cannot modify any components post-injection, it is reasonable to inject multiple backdoor behaviors during the backdoor training stage. Besides, the size of PTM also impacts attack performance as discussed in Section VI-E. Fig. 7 presents the impact of these factors on attack performance. We evaluate Transformer encoders of varying sizes (small: 0.6M, medium: 1.3M, and large: 2.3M parameters) with different numbers of trigger-POR pairs. The results reveal that our proposed backdoor

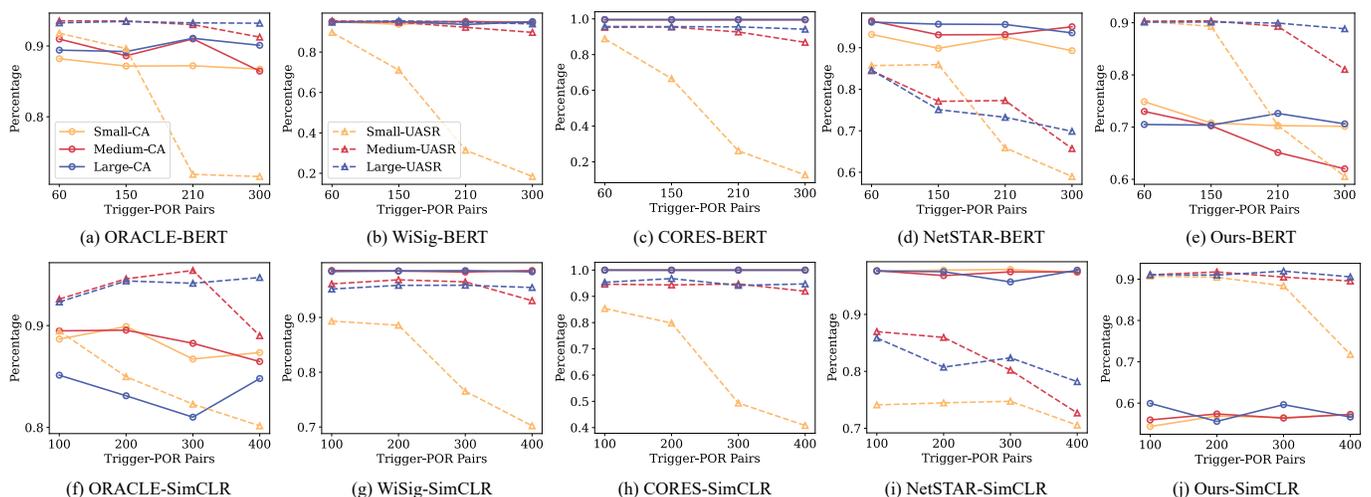


Fig. 7. Effects of PTM size and trigger-POR pairs on backdoor attacks in time domain BERT (top row) and time-frequency domain SimCLR (bottom row). Small-CA and Small-UASR denote the CA and UASR for small-sized PTMs.

attack generally achieves high CA and UASR across different configurations, indicating attack effectiveness. Compared to the small PTM, larger PTMs can maintain high CA and UASR in both the time domain and time-frequency domain. When increasing the number of trigger-POR pairs to implant more backdoor behaviors into PTMs, a clear trend emerges. Smaller PTMs experience drops in UASR, indicating they cannot retain a large number of backdoor behaviors while maintaining their utility. In contrast, larger PTMs can remember these backdoors and maintain high UASR. It is important to note that today’s foundation models continue to grow in size, becoming more capable of remembering backdoor behaviors while potentially offering stronger generalization performance compared to smaller models. This highlights a potential security concern in deploying PTMs in RF fingerprinting systems.

2) *PORs Design Comparison*: We evaluate the effectiveness of our proposed orthogonal PORs design by comparing it to the non-orthogonal PORs used in [20], which employs varying numbers of -1 s and 1 s. To ensure a fair comparison, we maintain consistency with our previous setup by using 8 trigger-POR pairs. In all cases, the CA is similar to ours, and the UASR only experiences drops in a few cases compared to our method. The most significant difference is observed in the TR metric as shown in Table VI. TR decreases in most cases using the non-orthogonal PORs design, with some cases achieving only 25%, indicating that their attack targets only two different downstream categories using 8 trigger-POR pairs. There are only four cases that can achieve the same TR as our orthogonal PORs method. Additionally, their method generates a constant number of PORs based on representation length, while ours can generate any number of orthogonal PORs. These results demonstrate that our orthogonal PORs design is crucial for successfully launching backdoor attacks on PTMs in a data-free setting. It allows for more effective targeting of multiple downstream categories, providing a more practical attack strategy for RF fingerprinting systems.

TABLE VI
PORs DESIGN COMPARISON. UNDERLINED VALUES INDICATE THE SAME TR AS OUR PROPOSED ATTACK.

SSL	Time Domain			Time-frequency Domain					
	SimCLR		TS-TCC	BERT	SimCLR		MAE		
Model	Res	Trans	Res	Trans	Trans	Res	Trans	Res	Trans
ORACLE	0.38	0.38	<u>0.50</u>	0.38	0.50	0.50	0.25	0.63	0.63
WiSig	<u>0.88</u>	0.38	0.63	0.25	<u>1.00</u>	0.25	0.25	0.50	0.50
CORES	0.63	0.38	0.63	0.25	0.38	0.38	0.25	0.50	0.63
NetSTAR	0.50	0.25	0.75	0.38	0.38	0.38	0.38	0.50	0.38
Ours	0.25	0.38	0.25	<u>0.38</u>	0.38	0.25	0.25	0.50	0.25

VII. CONCLUSION

In this paper, we propose the first protocol-agnostic and data-free backdoor attack on PTMs used in RF fingerprinting systems. Unlike traditional backdoor attacks where attackers may possess data and label information, we inject backdoors into unsupervised PTMs without downstream knowledge or access to downstream training. To achieve this, we employ three key strategies: utilizing substitute datasets, designing trigger sets, and manipulating output representations to inject backdoor behaviors into the PTMs. Extensive experiments are conducted across Wi-Fi and LoRa, using five different datasets and two mainstream SSL methods in both the time and time-frequency domain. Through this comprehensive analysis, we demonstrate that our proposed data-free backdoor attack poses a practical threat to RF fingerprinting systems, highlighting the urgent need for robust security measures to mitigate such threats when deploying PTMs in the real world. The authors have provided public access to their code at github.com/Tianyaz97/rf_backdoor.

ACKNOWLEDGMENTS

This work is supported in part by the NSF (CNS-2415209, CNS-2321763, CNS-2317190, IIS-2306791, and CNS-2319343).

REFERENCES

- [1] Y. Zou, J. Zhu, X. Wang, and L. Hanzo, "A survey on wireless security: Technical challenges, recent advances, and future trends," *Proc. IEEE*, vol. 104, no. 9, pp. 1727–1765, 2016.
- [2] E. Perenda, S. Rajendran, G. Bovet, M. Zheleva, and S. Pollin, "Contrastive learning with self-reconstruction for channel-resilient modulation classification," in *Proc. IEEE Conf. Computer Communications (INFOCOM)*. IEEE, 2023, pp. 1–10.
- [3] Q. Xu, R. Zheng, W. Saad, and Z. Han, "Device fingerprinting in wireless networks: Challenges and opportunities," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 1, pp. 94–104, 2015.
- [4] J. Zhang, G. Shen, W. Saad, and K. Chowdhury, "Radio frequency fingerprint identification for device authentication in the internet of things," *IEEE Commun. Mag.*, 2023.
- [5] S. Riyaz, K. Sankhe, S. Ioannidis, and K. Chowdhury, "Deep learning convolutional neural networks for radio identification," *IEEE Commun. Mag.*, vol. 56, no. 9, pp. 146–152, 2018.
- [6] J. Zhang, R. Woods, M. Sandell, M. Valkama, A. Marshall, and J. Cavallaro, "Radio frequency fingerprint identification for narrowband systems, modelling and classification," *IEEE Trans. Inf. Forensics Security*, vol. 16, pp. 3974–3987, 2021.
- [7] L. Peng, A. Hu, J. Zhang, Y. Jiang, J. Yu, and Y. Yan, "Design of a hybrid RF fingerprint extraction and device classification scheme," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 349–360, 2018.
- [8] K. Sankhe, M. Belgiovine, F. Zhou, S. Riyaz, S. Ioannidis, and K. Chowdhury, "ORACLE: Optimized radio classification through convolutional neural networks," in *Proc. IEEE Conf. Computer Communications (INFOCOM)*. IEEE, 2019, pp. 370–378.
- [9] G. Shen, J. Zhang, A. Marshall, L. Peng, and X. Wang, "Radio frequency fingerprint identification for LoRa using spectrogram and CNN," in *Proc. IEEE Conf. Computer Communications (INFOCOM)*. IEEE, 2021, pp. 1–10.
- [10] A. Al-Shawabka, F. Restuccia, S. D'Oro, T. Jian, B. C. Rendon, N. Soltani, J. Dy, S. Ioannidis, K. Chowdhury, and T. Melodia, "Exposing the fingerprint: Dissecting the impact of the wireless channel on radio fingerprinting," in *Proc. IEEE Conf. Computer Communications (INFOCOM)*. IEEE, 2020, pp. 646–655.
- [11] T. Zhao, X. Wang, and S. Mao, "Cross-domain, scalable, and interpretable rf device fingerprinting," in *Proc. IEEE Conf. Computer Communications (INFOCOM)*. IEEE, 2024, pp. 2099–2108.
- [12] T. Zhao, N. Wang, S. Mao, and X. Wang, "Few-shot learning and data augmentation for cross-domain uav fingerprinting," in *Proceedings of the 30th Annual International Conference on Mobile Computing and Networking*, 2024, pp. 2389–2394.
- [13] H. Li, K. Gupta, C. Wang, N. Ghose, and B. Wang, "RadioNet: Robust deep-learning based radio fingerprinting," in *Proc. IEEE Conf. on Communications and Network Security (CNS)*. IEEE, 2022, pp. 190–198.
- [14] Z. Chen, Z. Pang, W. Hou, H. Wen, M. Wen, R. Zhao, and T. Tang, "Cross-device radio frequency fingerprinting identification based on domain adaptation," *IEEE Trans. Consum. Electron.*, 2024.
- [15] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell *et al.*, "Language models are few-shot learners," *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.
- [16] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.
- [17] C. Liu, X. Fu, Y. Wang, L. Guo, Y. Liu, Y. Lin, H. Zhao, and G. Gui, "Overcoming data limitations: a few-shot specific emitter identification method using self-supervised learning and adversarial augmentation," *IEEE Trans. Inf. Forensics Security*, 2023.
- [18] J. Chen, W.-K. Wong, and B. Hamdaoui, "Unsupervised contrastive learning for robust RF device fingerprinting under time-domain shift," *arXiv preprint arXiv:2403.04036*, 2024.
- [19] J. Jia, Y. Liu, and N. Z. Gong, "Badencoder: Backdoor attacks to pre-trained encoders in self-supervised learning," in *IEEE Symp. on Security and Privacy (SP)*. IEEE, 2022, pp. 2043–2059.
- [20] L. Shen, S. Ji, X. Zhang, J. Li, J. Chen, J. Shi, C. Fang, J. Yin, and T. Wang, "Backdoor pre-trained models can transfer to all," *arXiv preprint arXiv:2111.00197*, 2021.
- [21] R. Ning, C. Xin, and H. Wu, "Trojanflow: A neural backdoor attack to deep learning-based network traffic classifiers," in *Proc. IEEE Conf. Computer Communications (INFOCOM)*. IEEE, 2022, pp. 1429–1438.
- [22] N. Carlini and A. Terzis, "Poisoning and backdooring contrastive learning," *arXiv preprint arXiv:2106.09667*, 2021.
- [23] A. Saha, A. Tejankar, S. A. Koohpayegani, and H. Pirsiavash, "Backdoor attacks on self-supervised learning," in *Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, 2022, pp. 13 337–13 346.
- [24] X. Liu, F. Zhang, Z. Hou, L. Mian, Z. Wang, J. Zhang, and J. Tang, "Self-supervised learning: Generative or contrastive," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 1, pp. 857–876, 2021.
- [25] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *International conference on machine learning*. PMLR, 2020, pp. 1597–1607.
- [26] M. Shao, P. Deng, D. Li, R. Lin, and H. Sun, "A specific emitter identification method based on self-supervised representation learning," in *2024 IEEE 4th Int. Conf. on Power, Electronics and Computer Applications (ICPECA)*. IEEE, 2024, pp. 125–128.
- [27] T. Zhao, X. Wang, J. Zhang, and S. Mao, "Explanation-guided backdoor attacks on model-agnostic rf fingerprinting," in *Proc. IEEE Conf. Computer Communications (INFOCOM)*. IEEE, 2024, pp. 221–230.
- [28] T. Zhao, J. Zhang, S. Mao, and X. Wang, "Explanation-guided backdoor attacks against model-agnostic rf fingerprinting systems," *IEEE Trans. Mobile Comput.*, 2024.
- [29] T. Zhao, Z. Tang, T. Zhang, H. Phan, Y. Wang, C. Shi, B. Yuan, and Y. Chen, "Stealthy backdoor attack on RF signal classification," in *Proc. IEEE Int. Conf. Computer Communications and Networks (ICCCN)*. IEEE, 2023, pp. 1–10.
- [30] T. Zheng and B. Li, "Poisoning attacks on deep learning based wireless traffic prediction," in *Proc. IEEE Conf. Computer Communications (INFOCOM)*. IEEE, 2022, pp. 660–669.
- [31] H. Xu, P. Zhou, R. Tan, M. Li, and G. Shen, "Limu-bert: Unleashing the potential of unlabeled data for imu sensing applications," in *Proc. of the 19th ACM Conf. on Embedded Networked Sensor Systems*, 2021, pp. 220–233.
- [32] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in *Proc. IEEE/CVF Conf. on computer vision and pattern recognition*, 2022, pp. 16 000–16 009.
- [33] H. Qian, T. Tian, and C. Miao, "What makes good contrastive learning on small-scale wearable-based tasks?" in *Proc. ACM SIGKDD Conf. on knowledge discovery and data mining*, 2022, pp. 3761–3771.
- [34] E. Eldele, M. Ragab, Z. Chen, M. Wu, C. K. Kwok, X. Li, and C. Guan, "Time-series representation learning via temporal and contextual contrasting," *arXiv preprint arXiv:2106.14112*, 2021.
- [35] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. on computer vision and pattern recognition*, 2016, pp. 770–778.
- [36] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.
- [37] S. Hanna, S. Karunaratne, and D. Cabric, "Open set wireless transmitter authorization: Deep learning approaches and dataset considerations," *IEEE Trans. on Cogn. Commun. Netw.*, vol. 7, no. 1, pp. 59–72, 2020.
- [38] D. Raychaudhuri, I. Seskar, M. Ott, S. Ganu, K. Ramachandran, H. Kremo, R. Siracusa, H. Liu, and M. Singh, "Overview of the ORBIT radio grid testbed for evaluation of next-generation wireless network protocols," in *Proc. IEEE Wireless Communications and Networking Conference*, vol. 3. IEEE, 2005, pp. 1664–1669.
- [39] S. Hanna, S. Karunaratne, and D. Cabric, "WiSig: A large-scale wifi signal dataset for receiver and channel agnostic RF fingerprinting," *IEEE Access*, vol. 10, pp. 22 808–22 818, 2022.
- [40] A. Elmaghbbub and B. Hamdaoui, "LoRa device fingerprinting in the wild: Disclosing RF data-driven fingerprint sensitivity to deployment variability," *IEEE Access*, vol. 9, pp. 142 893–142 909, 2021.
- [41] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation forest," in *Proc. 8th IEEE Int. Conf. Data Mining*. IEEE, 2008, pp. 413–422.
- [42] Y. Gao, C. Xu, D. Wang, S. Chen, D. C. Ranasinghe, and S. Nepal, "Strip: A defence against trojan attacks on deep neural networks," in *Proc. 35th Annual Computer Security Applications Conf.*, 2019, pp. 113–125.
- [43] K. Liu, B. Dolan-Gavitt, and S. Garg, "Fine-pruning: Defending against backdoor attacks on deep neural networks," in *Proc. Int. Symp. Research in Attacks, Intrusions, and Defenses*. Springer, 2018, pp. 273–294.