IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 15 November 2023, accepted 3 December 2023, date of publication 7 December 2023,
date of current version 18 December 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3340305

RESEARCH ARTICLE

Radio Frequency Public Key Generator for Digital Cryptographic Application

JUSUNG KANG[1], YOUNG-SIK KIM[2], (Member, IEEE), AND HEUNG-NO LEE[1], (Senior Member, IEEE)

[1]School of Electrical Engineering and Computer Science, Gwangju Institute of Science and Technology, Gwangju 61005, South Korea
[2]Department of Electrical Engineering and Computer Science, Daegu Gyeongbuk Institute of Science and Technology, Daegu 42988, South Korea

Corresponding author: Heung-No Lee (heungno@gist.ac.kr)

This work was supported by the Ministry of Science and ICT (MSIT), South Korea, through the Information Technology Research Center (ITRC) Support Program, supervised by the Institute for Information and Communications Technology Planning and Evaluation (IITP), under Grant IITP-2023-2021-0-01835.

ABSTRACT In Internet of Things (IoT) environments, effective public key management is crucial for managing numerous devices. RF features, primarily considered analog features within physical layer authentication by RF Fingerprinting (RFF) processes, present a novel approach to key management. In this research, we introduce a novel RF-based Public Key Generator (RF-PubKG) model that maps RF features into cryptographic sequences by incorporating a Key Generation (KeyGen) layer into the RFF model. The RF-PubKG demonstrates superior performance, achieving 97.2% accuracy at a 20dB SNR and further improving to 99.6% in noise-free conditions with a Frame Error Rate (FER) below 1%. The generated public key sets exhibit negligible correlation, with intra-key-set correlations not exceeding 0.24 and inter-model correlations falling below 0.04, highlighting the reliability of the RF-PubKG model. The integration with the Rivest–Shamir–Adleman (RSA) algorithm provides proof-of-concept for the RF-PubKG-based digital signature scheme, effectively simplifying the Certificate Authorities (CAs) management and, consequently, reducing Public Key Infrastructure (PKI) complexity. This simplification promises effective public key management within the Public Key Cryptography (PKC), thereby enhancing the efficiency of digital signature verification processes.

INDEX TERMS Radio frequency fingerprinting, public key generator, public key cryptography, digital signature scheme, public key infrastructure.

I. INTRODUCTION

In an Internet of Things (IoT) environment, accepting messages from trusted users is a crucial task. A common method to authorize users is to check the device's address, such as the Media Access Control (MAC) or Internet Protocol (IP) address, encoded as digital bits in message packets. However, if these addresses are transmitted without cryptographic encryption, eavesdroppers can sniff and modify the addresses using software-defined approaches [1].

One of the efficient solutions to solve this problem is to insert a valid authentication code in message packets. An overview of the user authentication scheme in IoT environments is presented in Fig. 1. In recent decades, several network security protocols, such as IEEE 802.1X [2], MACsec [3], or IPsec [4], have been proposed to secure network channels. These methods encrypt and decrypt user datagrams using cryptographic algorithms, making it impossible for eavesdroppers to sniff the user address without access to the public and private key details.

To secure communication between the numerous devices in IoT environments, a reliable key management system is essential. Public Key Cryptography (PKC) can be an effective solution due to its effective public key management structure. In PKC, the sender's datagrams are encrypted using a private key, allowing any receiver to verify both the integrity of the datagrams and the sender's identity. This approach simplifies key management, as each IoT device needs only a single

The associate editor coordinating the review of this manuscript and approving it for publication was Alessandro Pozzebon.

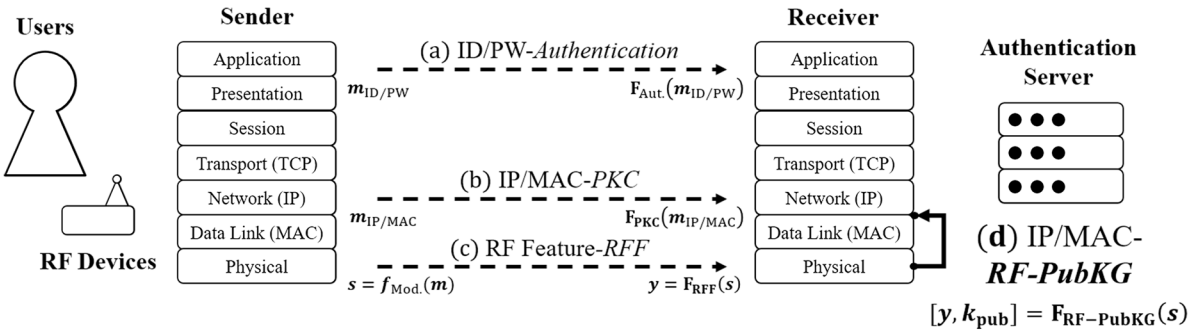

FIGURE 1. An overview of the user authentication scheme in IoT Environments: (a) ID/PW-based authentication; (b) authentication based on IP and MAC addresses utilizing PKC; (c) authentication using RF features based on RFF; and (d) (RF-PubKG) the proposed method for authentication using IP and MAC addresses combined with RF features through PKC.

public and private key, whereas a unique key for each pair of devices is required in private key cryptography.

Digital certificates are primarily used to verify the trustworthiness of the public key. These certificates are centrally managed in the Public Key Infrastructure (PKI) for key authentication, certificate issuance, and management [5]. However, establishing a trusted PKI involves significant financial and resource allocations for trusted third parties, which is not feasible for IoT environments. An alternative key management system for ensuring the trustworthiness of the public keys is required.

Radio Frequency Fingerprinting (RFF) can be an alternative approach for verifying the authenticity of IoT devices. The RFF is one of the physical layer authentication approaches that utilizes a unique RF feature present in analog RF signals. The inherent nonlinearity in the RF components of the transmitter, such as the Digital Analog Converter, Frequency Oscillator, or Power Amplifier, arises from manufacturing variations [6]. These effects accumulate and manifest as a distinct feature in the transmitted RF signal, which can serve as a unique authenticated key referred to as the RF feature.

Research on RF features is both extensive and multifaceted. For instance, the time-frequency energy properties of transient signals generated at the beginning of RF transmission have been used to identify twenty Bluetooth devices [7]. Multi-sampled steady state signals, capturing variations in RF transmission of preamble data, have been directly trained into a convolutional neural network for 54 ZigBee devices under line-of-sight conditions [8]. Spectrograms of falling signals observed during the decline of RF transmission have been used to identify seven frequency-hopping transmitters [9]. More recently, multifaceted RF features have been considered with advanced deep learning approaches. For Wi-Fi devices, IQ, carrier frequency offset, Fourier coefficients, and time-frequency coefficients are incorporated into an attention-based deep learning model [10]. Similarly, magnitude, phase, and power spectral density of steady-state signals of the Bluetooth devices have been considered with an embedding-attention framework [11].

The RF features are renowned for their non-replicable key characteristics, largely attributed to practical challenges [12]. The randomness and uniqueness of these RF features stem from the natural variations introduced during the manufacturing process. Replicating these key features would require tighter control over the varied components at the analog level. However, it is widely recognized that achieving such control is either prohibitively expensive or virtually impossible in real-world scenarios [13]. Owing to the inherent randomness and uniqueness of the RF features, they can be effectively utilized as non-replicable public keys in user authentication schemes.

Our research goal is to utilize the non-replicable RF features as public keys for the PKC. To achieve this goal, the RF feature must be converted into a finite cryptographic sequence. Recent research on RFF has concentrated on capturing RF features as digitized signals in the real domain [7−11, 14−16]. These features are segmented directly from the RF signal and transformed into feature domains to enhance the distinction between different RF transmitters. Subsequently, these RF features have mainly been processed using AI models for identification, rather than for cryptographic computations. Conversely, cryptographic schemes based on PKC depend on complex mathematical problems conducted within the finite field domain [17]. Although cryptographic calculation in the real domain is feasible, it requires hardware capabilities that are expensive and unsuitable for IoT environments. Therefore, further research is required to establish a mapping relationship between RF features and cryptographic sequences.

In this paper, we propose a novel RFF process for a Radio Frequency based Public Key Generator (RF-PubKG) to utilize RF features as cryptographic sequences. In the process of RFF model training, we introduce a key generation layer to map the RF features into cryptographic sequences. By considering these cryptographic sequences as users' public keys, we can simplify the PKI structure in the PKC, enhancing the

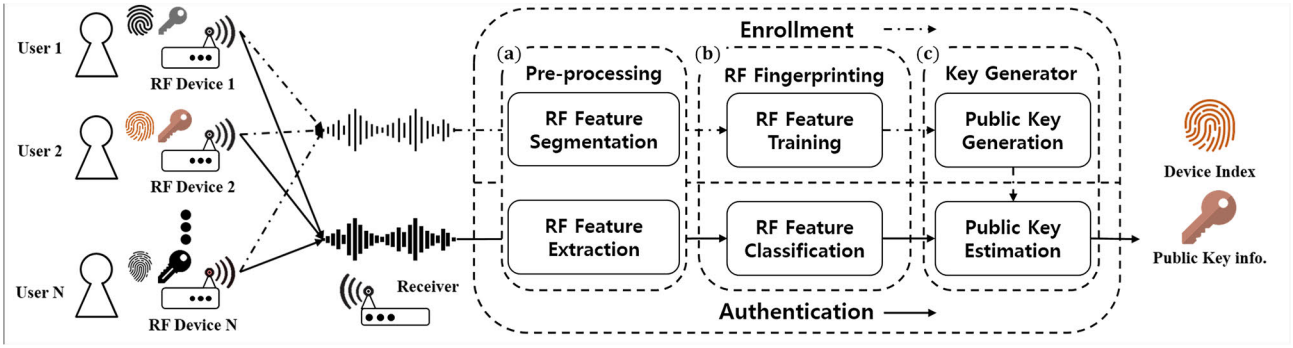

FIGURE 2. The overall RF Fingerprinting process is depicted with (a) the Pre-processing step and (b) the RF Fingerprinting step illustrating the conventional RFF, followed by (c) (Proposed) the Key Generator step representing the proposed RF-PubKG method.

efficiency of the public key management system. The specific contributions of this paper are detailed as follows:

- ✓ ***(RF-basedCryptographic sequences)*** We propose the RF-PubKG for mapping RF features to cryptographic sequences. This unique mapping of the RF feature to a cryptographic sequence enables integrated authentication with RFF and PKC. This aspect of the research paves the way for addressing cryptographic problems based on RF features.
- ✓ ***(RF-based Digital Signature***) As proof of concept, we evaluate the RF-PubKG-based digital signature scheme along with the hashed RSA algorithm. This result demonstrates the simplification of the PKI structure by relying on the trustworthiness of the public key, which is inferred from non-replicable RF features.

The structure of this paper is as follows: Chapter II describes the background knowledge related to RFF and the digital signature scheme for PKC. Chapter III presents the proposed RF-PubKG scheme and the conceptual structure for the RF-PubKG-based digital signature scheme implemented by RSA algorithm. Chapter IV details the dataset and experimental setup, while Chapter V presents the results and discussions. Chapter VI concludes the paper by summarizing the findings of this research.

II. BACKGROUND KNOWLEDGE

A. TARGET RADIO FREQUENCY FEATURES

The RF features are distinct characteristics that can be differentiated within the RF domain. Selecting the appropriate target RF feature is crucial for the design of the RFF. While various methods exist to calculate the RF features from RF signals, in this work, we adopt the definition outlined in our previous research [9]. The simple descriptions for each feature are as follows:

- ✓ ***(Rising Transient, RT)*** The RT feature is a signal property that rises from the noise level to the desired communication level, illustrating the process of RF signal emission when the device is activated.
- ✓ ***(Steady State, SS)*** The SS feature refers to the property of the signal region that contains the RF-modulated digital data for message transmission. This illustrates the process of RF modulation during data transmission.
- ✓ ***(Falling Transient, FT)*** The FT feature represents the signal property that decreases from the communication level to the noise level. This illustrates the attenuation of the RF signal when the device is deactivated.

The most crucial aspect of utilizing the RF features as public keys is ensuring that it cannot be forged by a third party. In [18], it was reported that the statistical analysis of the RT feature is more resilient to impersonation attacks compared to the I/Q constellation error calculation in the SS feature.

Additionally, we aim to replace certificates with RF features. From the definition of the RF features, the RT and FT features are independent of the modulated digital data, and they can be directly utilized as unique features over an extended period. On the other hand, the SS feature undergoes significant variations depending on the modulated digital data. These data dependencies can be eliminated through additional computational costs, such as extracting the ideal modulated RF signal from the received RF signal [19]. However, these post-processing costs may compromise the effectiveness of the system configuration.

For this reason, in this research, we focus on the RT and FT features as RF features, aiming to create characteristics that have no dependencies on the digital contents of the certificate.

B. RADIO FREQUENCY FINGERPRINTING

The overall scheme for the RFF process is presented in Fig. 2, along with the proposed RF-PubKG structure. This subsection describes the detailed RFF process, including the pre-processing step and the RFF model training process. The proposed RF-PubKG scheme is described in Chapter III.

The details of the RFF process are conducted in three steps: 1) RF feature segmentation, 2) RF feature extraction, and 3) RF feature training and classification.

The overall RFF process is formulated as a classification problem for given RF features. The mathematical description is as follows:

$$\mathbf{y} = \mathrm{F}_{\mathrm{RFF}}(\mathbf{s}) \tag{1}$$

where $\mathbf{s} \in \mathbb{C}^{N_{\text{Sig}} \times 1}$ is a down-converted RF signal acquired from the receiver operation. N_{Sig} is the length of a complex-valued signal $\mathbf{s}$. F_{RFF} is the RFF algorithm that maps the input signal $\mathbf{s}$ from the RF signal space to the device ID space. Finally, $\mathbf{y} \in \mathbb{R}^{N_{\text{C}} \times 1}$ is the result of the RFF that contains the device ID information, where N_{C} represents the number of transmitters used to train the RFF algorithm.

The RF feature segmentation is the step for extracting the target RF features from the received RF signal. This procedure can be represented by the following equation:

$$\mathbf{s}_{\text{Seg}} = \text{g}_{\text{Seg}}(\mathbf{s}) \tag{2}$$

where g_{Seg} is the function for segmenting the RF features $\mathbf{s}_{\text{Seg}} \in \mathbb{C}^{N_{\text{Seg}} \times 1}$. It is defined on the target RF feature list, i.e., *feature* $\in \{RT, FT\}$. N_{Seg} is the length of the segmented RF features $\mathbf{s}_{\text{Seg}}$. In this paper, g_{Seg} is designed based on the energy variation of the RF feature. We adopt a windowed energy detection approach to monitor the energy fluctuation, with $E_n \geq (1+\delta)E_{n-1}$ indicating a rise for RT and $E_n \leq (1+\delta)E_{n-1}$ indicating a fall for FT. The specific details are further described in [9].

As a next step, the RF feature extraction aims to transform the signal space of RF features into other domains, thereby enhancing the differentiation between the RF features from different transmitters. The extraction procedure is expressed as follows:

$$\mathbf{s}_{\text{Trans}} = h_{\text{Trans}}\left(\mathbf{s}_{\text{Seg}}\right) \tag{3}$$

where h_{Trans} is the function for domain transform of the RF features. $\mathbf{s}_{\text{Trans}} \in \mathbb{R}^{N^i_{\text{Trans}} \times N^j_{\text{Trans}}}$ is the transformed RF feature, where N^i_{Trans} and N^j_{Trans} are the sizes of each transformed indices, i and j, respectively. The function h_{Trans} can be defined in many different ways. It can transfer to the I/Q constellation domain [20], can calculate the properties in the statistics domain [21], or can be directly processed into the AI models for deep learning classifiers [22]. In this paper, the discrete-time short-time Fourier transform (STFT) is applied to convert the signal domain into multi-dimensional spaces, i.e., time and frequency axis. In this case, i and $j$are t and f; the details are described in [9].

As a last step, the RF feature training and classification aims to assign transmitter IDs from the RF features, ensuring robust classification through effective training. The classification results can be obtained by:

$$\mathbf{y} = \text{f}_{\text{Classify}}(\mathbf{s}_{\text{Trans}}) \tag{4}$$

where $\text{f}_{\text{Classify}}$ is the classification algorithm, and the output $\mathbf{y}$ implies the transmitter ID information. Thanks to recent research in deep learning, $\text{f}_{\text{Classify}}$ is commonly defined as a deep learning-based classifier, such as Convolutional Neural Network (CNN) based classifiers [23], [24], [25] or Generative Adversarial Network (GAN) based approaches [26], [27]. This paper utilizes the Deep Inception Network (DIN), which reported its effectiveness in understanding the RF features in our previous work [9], as the main RFF model. The structure details are described in Table 2 of Chapter IV.

To obtain the classification from (4), we must train the RFF model f_{Classify} for robustness. From the given training dataset $\mathbf{S} = [\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_M]$ of M samples and their relative labels $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_M]$, our DIN model for RFF can be trained with the cross-entropy loss and Adam optimizer [28] as follows:

$$loss = -\left(\frac{1}{M}\right)\sum_{i=1}^{M} \log\left(\frac{e^{\mathbf{y}_i[c_k]}}{\sum_{j=1}^{C} e^{\mathbf{y}_i[c_j]}}\right) \tag{5}$$

where k is the true transmitter ID relative to an output label $\mathbf{y}_i$, and $\mathbf{y}_i[c_j]$ is the value of the j th element in $\mathbf{y}_i$.

C. DIGITAL SIGNATURE SCHEME

A Digital Signature (DS) scheme is one of the PKC applications used to verify the authenticity and integrity of a digital message [29]. It involves using a private key to generate a unique signature for the sending message, which can be verified using the corresponding public key. The signature verifies that the message is not tampered with and is indeed sent by the claimed sender.

The DS scheme involves the following steps:

- ✓ ***(Key Generation, Gen)*** The signer generates a pair of keys; a private key, k_{pri}, and a corresponding public key, k_{pub}. k_{pri} is kept secret and used only by the signer, while k_{pub} is made public and can be shared with the verifier.

$$[k_{\text{pri}}, k_{\text{pub}}] = Gen(1^n) \tag{6}$$

- ✓ ***(Signature Generation, Sign)*** To sign the message m which has been hashed with the hash function h, the signer applies a one-way cryptographic function to the hashed message.

$$\sigma = Sign\left(k_{\text{pri}}, h(m)\right) \tag{7}$$

- ✓ ***(Signature Verification, Vrfy)*** The verifier with the public key, k_{pub}, and the signature, σ, can verify the authenticity of the sending message m. Verification is done by applying a verification function defined as follows:

$$Vrfy\left(k_{\text{pub}}, h(m), \sigma\right) = b \tag{8}$$

where b is 1 if the signature is valid, and 0 if the signature is invalid

The digital signature can be verified by anyone who has access to the valid public key. However, calculating the private key solely from the public key and forging a valid signature using the estimated private key is extremely challenging due to cryptographic complexity. These properties make digital signature schemes fundamental tools for ensuring the integrity of digital data and emphasize the need for valid public key management.

D. CERTIFICATES AND PUBLIC KEY INFRASTRUCTURE (PKI)

The trustworthiness of the public keys is crucial in digital signature schemes. If third parties generate invalid signatures

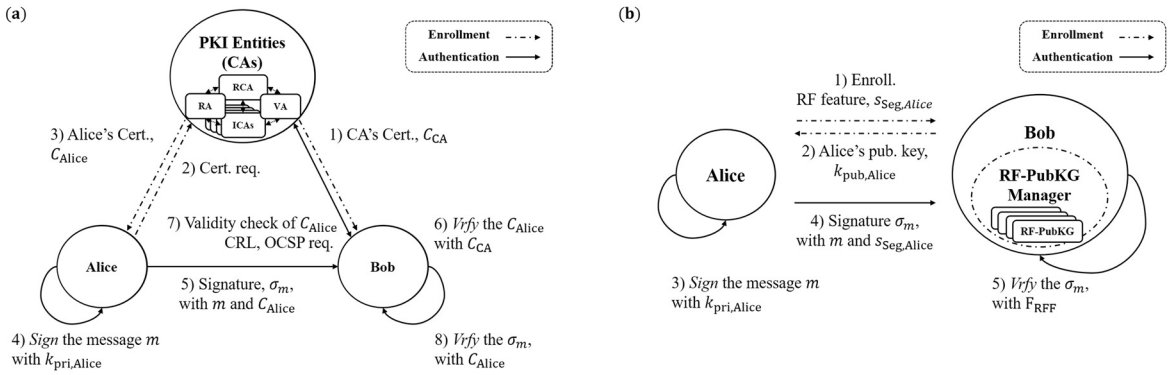

FIGURE 3. System overview of Digital Signature schemes: (a) Traditional scheme with a certificate management system from the PKIs, illustrating the process of certificate issuance; (b) Trustworthy scheme based on the RF-PubKGs, utilizing the RFF for key generation to eliminate the need for a centralized certificate authority, thus simplifying the overall certificate management system.

and fake public keys, the verifier may struggle to determine signature validity. To prevent this, the digital certificate is used to verify the authenticity of the user's public key. These certificates are strictly managed in the PKI system to ensure integrity and authenticity [30].

We present the system overview of the digital signature scheme with the PKIs in Fig. 3.a. The Certificate Authority (CA) is a core entity of the PKI structure, constructed as a trusted entity responsible for ensuring the authenticity of the certificates. The certificate contains a digital signature with a user's public key and identity information. The CA signs this certificate with the CA's private key and commits it to Alice and Bob within a secured channel. By verifying Alice's certificate with the CA's public key, Bob can trust the integrity of Alice's public key.

The CA is responsible for the revocation and renewal of the certificates. The CA must publish the Certificate Revocation Lists (CRLs) or operate the Online Certificate Status Protocol (OCSP) to inform users of the up-to-date status of the certificates. Bob needs to check these lists to ensure that Alice's public key is current.

The CAs are organized in a hierarchical model, where the intermediate CA (ICA) authenticates users, and the ICAs are authenticated by the root CA (RCA). This structure ensures the certificate's credibility by tracing back to the credibility of all CAs. However, this structure requires significant resource allocation to maintain the secure channel for the commitment of the certificates. Effective architecture to reduce these management costs must be considered [5].

In this paper, we aim to propose the RF-PubKG, which the public keys are directly derived from the RFF models.

$$[\mathbf{y}, k_{\text{pub}}] = \text{F}_{\text{RF-PubKG}}(\mathbf{s}) \tag{9}$$

We present an overview of the digital signature scheme with the RF-PubKG in Fig. 3.b. The uniqueness and non-replicability of RF features allow the RF features to serve as unique public keys, replacing the role of digital certificates. By transforming the RF features into finite cryptographic sequences, i.e., unique public keys, the trustworthiness of the public key can be ensured, a role previously fulfilled by certificates. This approach can simplify the PKI architecture in the digital signature scheme. It allows the RF-PubKG model manager, which originally operates at the receiver in the RFF process, to manage the enrollment of RF features for authentication. Therefore, this approach can simplify the hierarchical model required by focusing on managing the RFF models within the RF-PubKG manager, thus reducing the complexity of the traditional certificate infrastructure.

III. PROPOSED METHOD

A. RADIO FREQUENCY PUBLIC KEY GENERATOR

Originating from the conventional RFF process outlined in Chapter II-B, the RF-PubKG includes an additional feature map layer to enhance cryptographic key reliablity. The underlying principles and algorithmic approaches of the RF-PubKG are described in this section.

The feature map was first introduced as an intermediate computation result of the CNN model [31]. It is used to detect and extract specific features from the input data. Each value of the feature map in the data represents the interaction between a particular feature and its location information. This feature map helps to understand how the deep learning model can recognize essential features within the data.

From research analyzing the feature maps learned by each layer, it is well-known that higher layers can learn complex features to make decisions [32]. Applying this understanding to the RFF model, we can infer that the higher layers of the RFF model learn the crucial features from RF signals to estimate the device ID information. The proposed RF-PubKG is derived from this inference.

The RF-PubKG scheme is depicted in Fig. 4. Based on the definition of the RFF classification model in (4), we define the output of the Key Generation (KeyGen) Layer as follows:

$$[\mathbf{y}, k_{\text{raw}}] = \text{f}_{\text{Classify,KeyGen}}(\mathbf{s}_{\text{Trans}}) \tag{10}$$

where $k_{\text{raw}} \in \mathbb{R}^{N_{\text{key}} \times 1}$ is the raw key derived from the input RF feature $\mathbf{s}_{\text{Trans}}$, and is considered as an intermediate output produced by the additional KeyGen layer. $\text{f}_{\text{Classify,KeyGen}}$ is

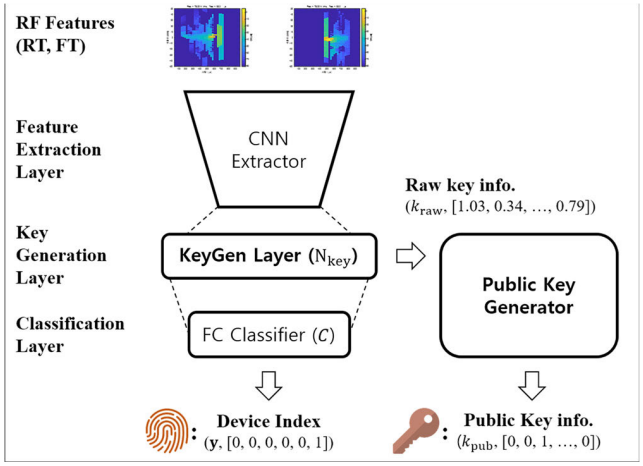

FIGURE 4. The proposed RF-PubKG structure: The KeyGen layer is located at the highest hidden layer, which processes outputs to generate and estimate the public key, as depicted in (10) to (14).

the classification algorithm for RF-PubKG, which extends the RFF model with the KeyGen layer. In this research, the KeyGen layer is located at the highest hidden layer of the RFF model, i.e., just before the final classification layer.

The raw key, k_{raw}, is computed within the real domain, $\mathbb{R}$. To apply the cryptographic scheme, the raw key needs to be converted into a cryptographic sequence that operates within the finite field domain. To transfer the real domain into the target finite field with q elements, we define a mapping function as follows:

$$k_{\text{estimate}} = \text{round}\left((q-1) \times \frac{k_{\text{raw}} - \min(k_{\text{raw}})}{\max(k_{\text{raw}} - \min(k_{\text{raw}}))}\right) \tag{11}$$

where $k_{\text{estimate}} \in \mathbb{N}^{N_{\text{key}} \times 1}$ is the estimated cryptographic key working with the Galois Fields with q elements, i.e., $GF(q)$. This research assumes a binary field, $GF(2)$. The round function is a mathematical function that maps a real number to the nearest integer number.

The phase of the RF-PubKG is divided into two parts, i.e., *Enrollment* and *Authentication,* similar to the RFF as described in Fig. 2. Enrollment is the training phase in which the classification model is trained from the pre-enrolled RF features of the target transmitters. Authentication is the testing phase in which the input RF feature is classified as one of the trained transmitter sets.

Based on this description, we set the public key of the RF transmitters as a sample mean of the pre-enrolled RF features, which can be obtained during the Enrollment phase. The detail is as follows:

$$k_{\text{pub},c_i} = \text{round}\left(\frac{\sum k_{\text{estimate,Train},c_i}}{n_{\text{Train},c_i}}\right) \tag{12}$$

where $k_{\text{pub},c_i} \in \mathbb{N}^{N_{\text{key}} \times 1}$ is the public key of the i th target device c_i, $k_{\text{estimate,Train},c_i}$ is the sample cryptographic key estimated from the pre-enrolled RF features, and n_{Train,c_i} is the number of the pre-enrolled samples.

The public key setting step can be done by calculating the public keys as in (12) for all of the target transmitters.

$$\mathbf{K}_{\text{Pub}} = \left[k_{\text{pub},c_1}, k_{\text{pub},c_2}, \ldots, k_{\text{pub},c_N}\right] \tag{13}$$

where $\mathbf{K}_{\text{Pub}} \in \mathbb{N}^{N_{\text{key}} \times N_C}$ is the public key set of all target transmitters, which can be used as a reference for the public key generators.

To authenticate the public key from the input RF feature during the Authentication phase, a key estimation method is required to estimate a public key from a given public key set. We consider the similarity of the cryptographic sequences; Hamming distance is a valuable metric that measures the similarity of the two input sequences [33]. We can estimate the public key as follows:

$$\hat{k}_{\text{pub}} = \underset{k_{\text{pub},c_i}}{\text{Argmin}}\, H(k_{\text{pub},c_i}, k_{\text{estimate,Test}}) \tag{14}$$

where $k_{\text{estimate,Test}}$ is the cryptographic key estimated from the test RF feature, H is the Hamming distance between the public key and estimated key, and $\hat{k}_{\text{pub}} \in \mathbb{N}^{N_{\text{key}} \times 1}$ is the final estimated public key of the test RF feature during the Authentication phase.

By referencing (10) and (14), we can obtain the formulation of the RF-PubKG as represented by (9). The whole procedure for the RF-PubKG is presented in Algorithm 1.

Algorithm 1 Proposed RF-PubKG algorithm, $\text{F}_{\text{RF-PubKG}}$.

Input: The received RF signal **s**.
  Step1: Segment and Transform the target RF signal **s** to the segmented RF feature $\mathbf{s}_{\text{Seg}}$ on (2) and the transformed RF feature $\mathbf{s}_{\text{Trans}}$ on (3).
If *phase* is *Enrollment* **do:**
  Step 2-1: Train the RF-PubKG model $\text{f}_{\text{Classify,KeyGen}}$ on (10) with the *loss* function on (5)
  Step 2-2: Set the public keys $\mathbf{K}_{\text{Pub}}$ on (12) and (13).
else if *phase* is *Authentication* **do:**
  Step 2-1: Estimate the device ID, c_i from the model output **y**, described in (10).
  Step 2-2: Estimate the public key, $\hat{k}_{\text{pub}}$, based on the key estimation equation in (14).
Output: The estimated device ID, c_i, the estimated public key, $\hat{k}_{\text{pub}}$.

B. RF-PUBKG BASED HASHED RSA SCHEME

In this chapter, we aim to prove the effectiveness of the proposed RF-PubKG system. As a proof of concept, we demonstrate the RF-PubKG-based digital signature scheme with the simplified PKI configuration by replacing the CAs with the RF-PubKGs.

This paper considers the hashed RSA algorithm as a digital signature scheme. The RSA algorithm is currently the most widely used PKC algorithm. The RSA is based on the mathematical fact that the factorization of the sufficiently large number is difficult to solve [34]. T, represented as follows:

$$n = P \cdot Q \tag{15}$$

Algorithm 2 Hashed RSA algorithm based on RF-PubKG.

Input: The public key k_{pub}, the user identity m, the received RF signal $\mathbf{s}$ and the RF-PubKG algorithm $\text{F}_{\text{RF-PubKG}}$.

function *Gen*(k_{pub})

1. Set the LSB of the k_{pub} as 1 (for odd number)
2. Set the large P and Q as prime numbers (with the size of $N_{\text{key}}/2$ bits).
3. Compute $n = P \cdot Q$ and $\varphi(n) = (P-1) \cdot (Q-1)$.
4. Check that gcd $\left(\varphi(n), k_{\text{pub}}\right) = 1$
   4.1 If not, do again from 2 to 4.
5. Compute k_{pri} where $k_{\text{pri}} \cdot k_{\text{pub}} \equiv 1 (\text{mod} \varphi(n))$

Output: Public key $\{k_{\text{pub}}, n\}$ and Private key $\{k_{\text{pri}}, n\}$.

function *Sign*(k_{pri}, m)

1. Hash the input message, m, i.e. $\hat{m} \leftarrow SHA_{256}(m)$
2. Sign the hashed message $\hat{m}$, i.e. $\sigma_m \leftarrow \hat{m}^{k_{\text{pri}}} \bmod n$

Output: The signature σ_m of the message m.

function *Vrfy*($\mathbf{s}, m, \sigma_m$)

1. Estimate the public key, $\hat{k}_{\text{pub}}$, from $\text{F}_{\text{RF-PubKG}}$ as depicted in Algorithm 1

if $SHA_{256}(m) = \sigma_m^{\hat{k}_{\text{pub}}} \bmod n$ **then**
  return True
else:
  return False

where P and Q are prime numbers, and n is the product of these two primes. The factorization of the sufficiently large n into the unknown prime numbers, P and Q, is a complicated problem. However, if one of the two primes is known, calculating the other remaining prime becomes an easy problem.

The RSA key generation process utilizes the above relationship to generate the public and private key pair. In this work, we aim to generate an RSA private key, k_{pri}, satisfying the following two conditions when the estimated public key, k_{pub}, is given from Algorithm 1.

$$\text{gcd}\left(\varphi(n), k_{\text{pub}}\right) = 1 \tag{16}$$

$$k_{\text{pri}} \cdot k_{\text{pub}} \equiv 1 (\text{mod} \varphi(n)) \tag{17}$$

where $\varphi(n) := (P-1) \cdot (Q-1)$ is Euler's totient function of n in (13).

We detail the hashed RSA algorithm based on the RF-PubKG in Algorithm 2. To integrate the RF-based estimated public key into the RSA algorithm, the following two modifications are made to the conventional hashed-RSA algorithm:

$$\text{LSB}\left(k_{\text{pub}}\right) = 1 \tag{18}$$

$$\hat{k}_{\text{pub}} = \text{F}_{\text{RF-PubKG}}(\mathbf{s}) \tag{19}$$

where (18) reflects the public key for RSA key pairs that need to be odd numbers, and (19) is the expected public key that should be utilized in the verification step.

The remaining steps align with the standard RSA algorithm; the signer signs the message m with its private key k_{pri}, denoted by RSA signature σ_m, and the verifier verifies the signature σ_m with its public key k_{pub} in verification scheme. The 'Hash-and-Sign' paradigm, referred to as hashed RSA algorithm [29], employs a one-way hash function h to convert variable-length input message m to a fixed-length hash value $\hat{m}$. This conversion is computationally challenging to reverse, making it useful for constructing efficient and secure signatures. In this paper, SHA-256 is applied, which is well known to provide random oracle properties [35]. The system structure is presented in Fig. 3.b.

TABLE 1. RF feature dataset.

Class	Model Type	# of Signal Acquisitions	# of RF bursts
Device 1	Model 1	35 times	665
Device 2	Model 1		665
Device 3	Model 1		665
Device 4	Model 1		665
Device 5	Model 2		661
Device 6	Model 2		661
Total classes	6	Total bursts	3982

IV. EXPERIMENTAL SETUPS

A. RF FEATURE DATASET DESCRIPTION

We collected a set of RF signals from real RF transmitters. Six Ultra High Frequency (UHF) walkie-talkie transmitters were prepared; four were the SL1M Motorola, and two were the BD358 Hytera. All transmitters adhered to the Digital Mobile Radio (DMR) standard [36], which followed the two-slot Time-Division Multiple Access (TDMA) and four-level Frequency Shift Keying (4FSK) modulation protocol. The RF signal consisted of repeated RF bursts. These bursts occurred at intervals of 30ms. Each RF burst was constructed from the RT, SS, and FT features described in Chapter II-A. We considered the RT and FT features as the target RF features in this research.

The details of the RF feature dataset are presented in Table 1. An average of 664 RF bursts were measured for each transmitter, resulting in 3982 bursts from six transmitters. The RF dataset was divided into training and testing datasets at a ratio of 7:3, meaning that 2790 bursts were used for training, while 1192 bursts were reserved for testing.

B. HARDWARE AND SOFTWARE CONFIGURATIONS

We collected the RF signal using our RF receiver system. The system configuration is depicted in Fig. 5. The RF signal was transmitted to the receiver at a carrier frequency of 444.025MHz. This signal was then received and down-converted to a 1MHz IF signal, utilizing hardware components with the Nagoya NL-R2 UHF antenna, XL-11-411 RF mixer, and E4438C ESG vector signal generator, which functioned as the frequency oscillator. The IF signal was sampled at 20MHz using the PX14400 digitizer. We applied software-defined Digital Signal Processing (DSP) techniques to extract the RF burst from the IF signal. The energy detection approach described in [9] was applied to detect the RF

TABLE 2. Architecture of the RF-PubKG model (Based on DIN model in [9]).

Type	Filter size / Stride / padding	Output Shape
Input signal	-	205x124x1
Conv_1	3x3 / 2 / 0	102x61x32
Conv_2	3x3 / 1 / 0	100x59x32
Conv_3	3x3 / 1 / 1	100x59x32
Max Pool	3x3 / 2 / 0	49x29x32
2 x Inception	Inception-A [$N_F = 32$]	49x29x128
1 x Reduction	Reduction-A [$N_F = 32$]	24x14x192
2 x Inception	Inception-A [$N_F = 64$]	24x14x256
1 x Reduction	Reduction-A [$N_F = 64$]	11x6x384
Avg. Pool	Adaptive Avg. Pooling	384
Key Gen Layer	***logits***	N_{key}
Linear	logits	6

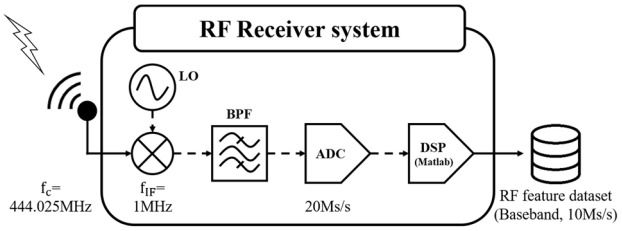

FIGURE 5. The RF receiver system: H/W and S/W configurations.

bursts, and this RF burst was subsequently down-converted to the baseband with a decimation factor of 2. Finally, we set the 10 M sampled baseband RF bursts as the RF Feature dataset.

All experiments were conducted on an Intel i7-6850K CPU and NVIDIA Titan RTX GPU, using Python 3.6 with PyTorch 1.6.0. The only exception was the DSP procedure for constructing the RF feature dataset, which was performed in MATLAB 2018a. The experiments were evaluated 10 times, and the average results are presented.

C. EVALUATED RF FINGERPRINTING MODELS

In this paper, we aim to demonstrate the effectiveness of the key generation approach rather than evaluating the classifier model. We describe the architectures of the main and baseline RFF models evaluated in this paper.

There are three approaches to constructing the custom deep learning classifier: a vgg block in VGG [37], a residual block in ResNet [38], and an inception block in Inception-v4 [39]. We have evaluated the RFF models with these construction approaches to demonstrate the generality of the RF-PubKG.

We constructed the main RFF model based on the inception block. The architecture detail is presented in Table 2. The design strategy of the inception block is to filter out input features using different receptive field sizes. This strategy was successfully demonstrated to be useful for understanding RF features in our previous work [9]. Based on this result, we adopted the DIN classifier from [9] as the main RFF model, utilizing the inception-A and reduction-A blocks of the inception-v4 model [39]. The only modification made was the introduction of the KeyGen layer as the highest hidden layer of the model, serving as the public key generator, as described in Chapter III-A.

TABLE 3. Key estimation accuracy*.

RF-PubKG models		Key Size 1024	2048	4096	8192
[P] Incep.	RT	98.3±0.2	98.2±0.4	98.2±0.4	97.9±0.8
	FT	96.0±0.7	95.1±1.0	94.9±0.8	93.7±1.7
	Ensem.	***99.7±1.0***	***99.5±0.2***	***99.6±0.3***	***99.4±0.4***
[B1] VGG	RT	92.0±4.1	95.7±1.1	96.7±1.2	97.0±0.5
	FT	84.5±1.8	84.4±1.8	85.8±1.3	85.8±1.0
	Ensem.	97.2±0.7	97.6±1.1	97.6±1.0	97.9±0.8
[B2] Res.	RT	97.1±0.9	97.2±0.3	96.5±0.8	96.0±0.6
	FT	91.2±0.8	91.5±1.1	90.6±1.2	90.8±1.2
	Ensem.	99.1±0.4	99.0±0.4	98.4±1.0	99.1±0.4
[B3] CNN	RT	65.6±11.7	73.2±7.7	79.2±4.4	81.9±5.4
	FT	78.3±2.1	76.6±6.4	78.6±3.3	78.5±3.6
	Ensem.	82.3±5.8	86.4±5.1	90.1±3.0	91.4±2.4

** Mean Accuracy (%) ± Standard Deviation, as derived from (14)*

A first baseline model is established using a set of vgg blocks. The design strategy of the vgg block employs a repeated pattern of a simple and homogeneous topology, proven effective in extracting complex features as the network deepens [37]. To ensure fairness in comparison with the main model, we simplified the VGG11 model in [37], originally composed of 5 vgg blocks, to just 2 vgg blocks with channel depths of 32 and 64. Similar to the main model, we introduced the KeyGen layer just before the FC-1000 layer.

A second baseline model is established using the residual block. The residual block is designed to alleviate the vanishing gradient problem that occurs as the network goes deeper, through the use of a skip connection [38]. This strategy is well-reflected in [40], where the Hilbert spectrum of the SS feature is effectively trained by repeating 2 residual blocks. Baseline 2 is constructed from the RFF model structure in [40] by introducing the KeyGen layer just before the fc5 layer.

The third baseline represents a conventional RFF model constructed using a CNN architecture. In [41], the SS features were calculated by removing the ideally encoded signal from the received RF signal, and these features were learned using an RFF model constructed with a 1D convolutional network. For the Baseline 3 model, we adapted the identification network from [41], converting the 1D convolutional layer to 2D, and added a KeyGen layer just before the final Dense layer.

D. ENSEMBLE RF-PUBKG

An ensemble approach is a well-known method to enhance the generalization performance of the classifier [42]. It aggregates the results of the multiple base classifiers to make a final decision. It was reported that the stacking ensemble of the multiple RF features can improve the accuracy of RFF [9].

We construct a stacking ensemble key generator as shown in Fig. 6. From the definition of the raw key with a single

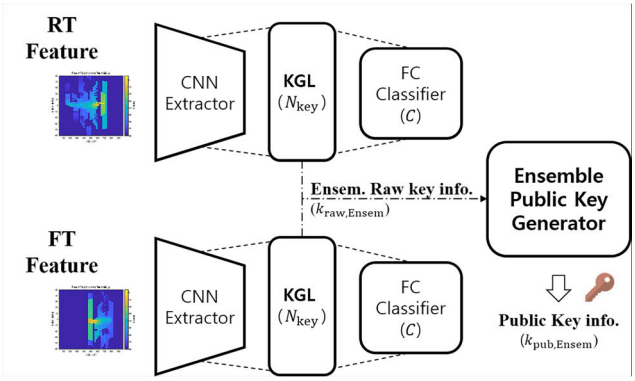

FIGURE 6. Ensemble approach of RF-PubKGs: The raw key outputs from each RF-PubKG are combined in stacked manner in (20).

classifier in (10), the ensemble raw key $k_{\text{raw,Ensem}}$ for the input RF features $\mathbf{s}_{\text{Trans,feature}}$ can be defined as follows:

$$k_{\text{raw,Ensem}} = \frac{1}{|feature|} \sum \text{f}_{\text{KeyGen,Classify}} \left(\mathbf{s}_{\text{Trans,feature}}\right) \quad (20)$$

where $feature \in \{RT, FT\}$ is the target feature of interest for stacking the ensemble classifier, and the other key generation algorithms in (10) to (14) operate on this ensemble raw key $k_{\text{raw,Ensem}}$.

V. RESULTS AND DISCUSSION

This chapter describes the results and discussion related to the proposed RF-PubKG method. Accuracy and Frame Error Rate (FER) are examined across various signal-to-noise radio (SNR) channel conditions to evaluate the effectiveness of the RF-PubKG. To evaluate system reliability, we measure clustering results for RF-PubKG outputs and the correlation between different public keys or RF-PubKG models. Furthermore, we quantify the size and time consumption of the RSA algorithm based on RF-PubKG, serving as a proof of concept for the RF-PubKG-based digital signature scheme, as illustrated in Fig. 3. These metrics, which will be discussed in the following subsections, provide a comprehensive evaluation of the proposed RF-PubKG's performance and reliability.

A. PUBLIC KEY ESTIMATION RESULTS

First, we evaluate the key estimation accuracy and key estimation time of the proposed RF-PubKG method. Accuracy is determined by the correctness of the estimated public key, as defined in (14). Time is measured as the public key generation time from (10) to (14). We assume that the public key set in (13) is already established during the Enrollment phase and committed to the receiver in the Authentication phase. The results are presented in Tables 3 and 4.

Table 3 illustrates the public key estimation accuracy related to variations in key size. The FT feature achieved an average mapping accuracy of 94.9% to cryptographic sequences, while the RT feature achieved 98.1% accuracy. The Ensemble of RF Features yielded a 99.6% accuracy in mapping to cryptographic public keys. This result indicates that there were just five rejections out of 1192 RF bursts in the test dataset. In other words, one rejection per every 7.5 seconds will occur in the DMR transaction.

TABLE 4. Key estimation time.

RF-PubKG models		Estimation Time*	# of parameters	# of branches
[P] Incep.	RT / FT	5.6 ± 0.1	1.1 M	4
	Ensem.	10.8 ± 0.1	2.3 M	
[B1] VGG	RT / FT	1.7 ± 0.0	25.8 M	1
	Ensem.	2.6 ± 0.1	51.5 M	
[B2] Res.	RT / FT	2.2 ± 0.0	0.3 M	2
	Ensem.	3.8 ± 0.0	0.5 M	
[B3] CNN	RT / FT	1.5 ± 0.0	0.8 M	1
	Ensem.	2.2 ± 0.0	1.7 M	

* *Mean Estimation Time (ms) ± Standard Deviation, as measured from*

When compared to the baselines, Baseline 2 achieved an accuracy that was 0.7% lower than the inception block, highlighting the residual block's efficiency. However, the inception block maintained higher accuracy than Baseline 2 across all key size variations. This is consistent with the findings in [9] that the consideration of different receptive filter sizes in the inception block is more efficient. While Baseline 1, utilizing the vgg blocks, showed similar estimation efficiency, it had a nearly 2% decrease in performance. Regarding Baseline 3, the CNN-based RFF model, it achieved an accuracy of only 87.6%. We analyzed this result due to its limited structure for training RT and FT features.

Table 4 presents the key estimation time for the RF feature-based public key. Using a single feature, the inception RF-PubKG results in an average estimation time of 5.6ms, while the ensemble approach results in 10.8ms. These values are greater than those of the other baselines, such as 2.2 ms for Baseline 3, 2.6 ms for Baseline 1, and 3.8 ms for Baseline 2.

Upon analysis, we observe that the time degradation is primarily correlated with the number of branches in RFF models, rather than with the number of parameters. For instance, the ResNet and inception blocks contain 2 and 4 branches, respectively. While these branches may appear to be calculated in parallel within the system structure, they are serially computed and combined in S/W implementations. This means that the inception blocks require 4 times as many calculations as other baselines. Even in that case, the results of the inception RF-PubKGs remain competitive, considering the one-slot duration of the DMR transaction is 30ms. We expect that the ensemble RF-PubKGs can be optimized in time by utilizing multiple GPU units for parallel input features.

As a next step, we estimate the key estimation accuracy against SNR variation according to the AWGN channel. From (1), the received signal, including the AWGN channel noise, is defined as follows:

$$\hat{\mathbf{s}} \leftarrow \mathbf{s} + \mathbf{n} \quad (21)$$

where $\hat{\mathbf{s}}$ is a noisy RF burst to which AWGN channel noise $\mathbf{n}$ generated proportionally from normal RF burst $\mathbf{s}$ is applied.

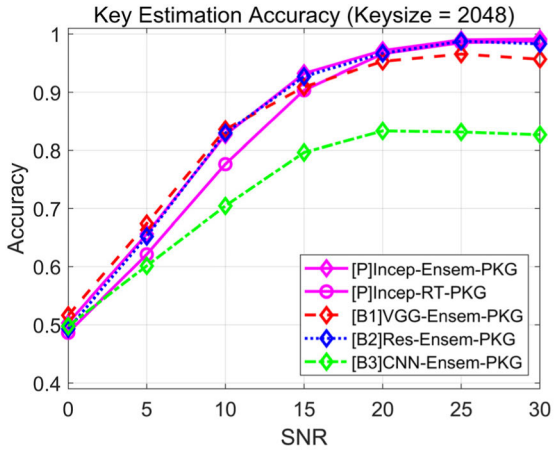

FIGURE 7. Key estimation accuracy of the RF-PubKG under AWGN channel conditions. The RF-PubKG achieves 97.2% at 20dB SNR, rising to 99.0% with improved channel conditions.

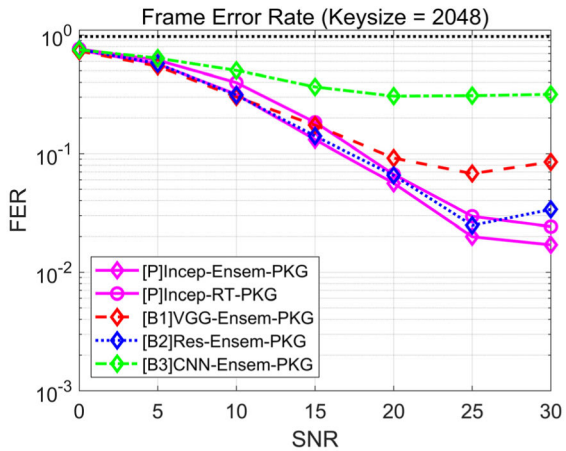

FIGURE 8. Frame error rate of the RF-PubKG under AWGN channel conditions. The RF-PubKG achieves 5.6% at 20dB SNR, 2.0% at 25dB SNR, and decreased to less than 1.0% in noise-free conditions.

SNR formulation for the AWGN noise $\mathbf{n}$ is defined as follows:

$$SNR = 10\log_{10}\left(\frac{\|\mathbf{s}\|_2^2}{|\mathbf{n}|\,\sigma_{\mathbf{n}}^2}\right) \tag{22}$$

where $|\mathbf{n}|$ represents the length of the $\mathbf{n}$, and $\sigma_{\mathbf{n}}^2$ represents its variation.

As a next evaluation metric, we evaluate the FER of the proposed method. According to the frame definition in the TDMA protocol of the DMR standard [36], one frame consists of two RF burst signals. Since the RF-PubKGs operate on units of the RF burst signal, we can compute the probability of two RF bursts being received without error. The probabilities are defined as follows:

$$BER = 1 - Accuracy \tag{23}$$

$$FER = (1 - BER)^2 \tag{24}$$

where the Burst Error Rate (BER) is the probability of an RF burst being rejected.

The estimation accuracy results in relation to SNR variation are presented in Fig. 7. With SNR over 20dB, which is generally assumed to be a good channel condition, the ensemble method achieved over 97.2% key estimation accuracy. This represents a performance improvement of more than 0.5% compared to 96.6% of the RT, 95.3% of the Baseline 1, and 96.7% of the Baseline 2. Baseline 3 only achieved an 83.3% accuracy, a degradation in performance. Especially at 25dB or higher, the inception RF-PubKGs achieved over 99.0% accuracy, uniquely achieving a BER of less than 1% compared to the other baselines.

Fig. 8 presents the FER results. At SNR levels exceeding 20dB, the ensemble approach achieves an FER of 5.6%. This value decreases to 2.0% at 25dB SNR and drops further to less than 1.0% in noise-free conditions where no AWGN noise is added. Conversely, the baselines do not reach below 1.0% FER, with the lowest value being 1.9% for Baseline 2 in noise-free conditions. We emphasize that these results reflect the raw FER without the application of Error Correction Coding (ECC), a technique commonly utilized to enhance FER performance. We anticipate that future improvements in performance through ECC will be possible.

B. RELIABILITY OF THE CRYPTOGRAPHIC SEQUENCES

To evaluate the reliability of the proposed RF-PubKG, we conduct a comparative analysis of the estimated public keys derived from both training and test datasets. To facilitate this comparison, we apply t-Distributed Stochastic Neighbor Embedding (t-SNE) to our RF dataset. t-SNE is a nonlinear dimensionality reduction method to transfer a high-dimensional data structure into a lower-dimensional space while preserving the similarity relationships of the data points [43]. It is primarily used for visualization and can be useful for discovering patterns or clusters in complex data domains. The results of the t-SNE are presented in Fig. 9.

Fig. 9.a. illustrates the clustering results between the public key sets $\mathbf{K}_{\text{Pub}}$ and the public keys k_{estimate} estimated from the training dataset. The results show that the public key estimation scheme in (12) is simple but effectively identifies the center of each cluster. The key consideration for this evaluation is how well this pre-enrolled public key set aligns with the public keys estimated from the test dataset. The clustering result is illustrated in Fig 9. b. The result shows that the given public key set retains the centrality of the clusters in the test dataset. It confirms that, as described in (14), accurate and unique public key estimation is achievable through a Hamming distance-based estimation approach, when using the provided public key set.

In Fig. 10, a correlation matrix of the public key sets is calculated to confirm the stability of the generated public cryptographic key sets. The result shows that the correlation between the generated Public Key of each transmitter is not significantly large, and the largest correlation is 0.24 between Tx1 and Tx3. This is a reasonable result considering that the AI model trains the dataset for optimizing the clusters with

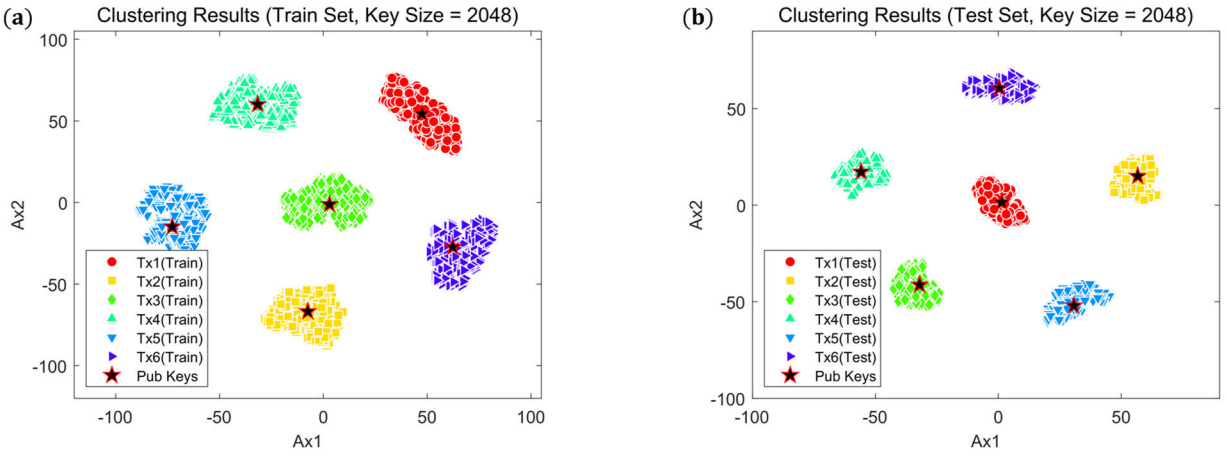

FIGURE 9. Clustering results of estimated public keys, k_{estimate}, from RF features with the public key set K_{Pub}: (a) Demonstrated centrality within the training dataset; (b) Consistency maintained within the testing dataset. The public key set accurately establishes cluster centers during training and preserves center integrity in testing.

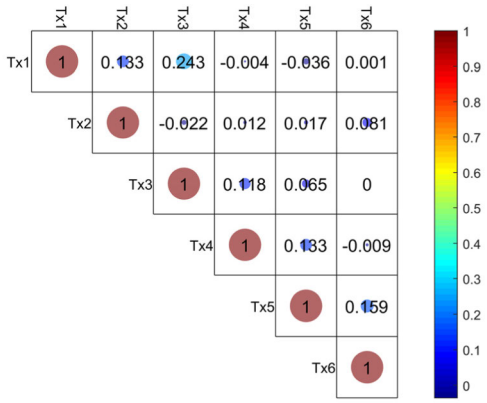

FIGURE 10. Correlation Matrix of the RF-PubKG. Public key correlations remain below 0.24, indicating the uniqueness of the generated public keys among RF transmitters.

TABLE 5. Correlation matrix for key sets generated by distinct RF-PubKG models.

RFF Models	Classes					
	Tx 1	Tx 2	Tx 3	Tx 4	Tx 5	Tx 6
Trial 1	1	1	1	1	1	1
Trial 2	0.01	-0.03	0.03	-0.01	-0.03	-0.02
Trial 3	*0.00**	0.02	0.01	0.02	0.02	*0.00**
Trial 4	*0.00**	-0.04	*0.00**	0.04	0.04	0.02
Trial 5	-0.02	*0.00**	*0.00**	-0.02	0.02	-0.01
Trial 6	-0.01	0.01	-0.01	*0.00**	*0.00**	0.01
Trial 7	0.00*	0.01	-0.04	-0.02	-0.02	-0.0.2
Trial 8	0.02	-0.01	*0.00**	*0.00**	*0.00**	0.00*
Trial 9	-0.03	0.04	-0.02	-0.01	-0.01	0.03
Trial 10	-0.03	0.01	0.01	-0.02	-0.02	-0.03

* *Correlation values are lower than 0.01*

sufficient distance. This result confirms that the RF-PubKGs can generate unique public key sets with sufficiently different cryptographic sequences between the RF transmitters.

Another significant aspect of evaluating the reliability of the key generator is to examine the variance in generated public key sets when new RFF models are being trained. Table 5 evaluates the correlations for the generated public key sets across the different trained RFF models. The result shows that the correlation remains consistently low, not exceeding 0.04. This implies that the activated node positions in the KeyGen layer are established through the random distribution. This observation confirms that the periodical re-training approach of the RFF model can enhance the overall security of the PKC system.

C. PUBKGS IN HASHED RSA SCHEME

As a proof of concept for the RF feature-based digital signature schemes, the implementation performance was evaluated using a hashed RSA algorithm based on the RF-PubKG. The results for size and time consumption are presented in Table 6. We implemented a hashed RSA digital signature scheme based on the X509 certificate for the PKI management system using the PyCryptodome [44] and pyOpenSSL [45] libraries. The system overview is depicted in Fig. 3. PyCryptodome, a Python library for cryptographic operations that complies with the Digital Signature Standard (DSS) standard documents NIST FIPS 186-4 [46], is utilized to implement the RSA signature scheme. Meanwhile, the pyOpenSSL library, a wrapper for OpenSSL in Python, is used to construct the digital signature scheme with a single CA for PKIs using the X509 object packages in pyOpenSSL. The evaluation included evaluations of certificate file sizes and scheme operation time, thus confirming the concept for the proposed RF-PubKGs.

In our evaluation, we assume that the training procedure for the RF-PubKG has already completed the Enrollment phase as defined in Algorithm 1. This implies that the RF-PubKG function F_{RFF} and Public Key Set $\mathbf{K}_{\text{Pub}}$ have been committed

TABLE 6. Quantification analysis of RF-PubKG based digital signature scheme implemented by hashed RSA algorithm.

Model	Key Size	Cert. File Size (Bytes)		Cert. Gen. time (ms)		Digital Signature processing time (ms)				
		CA	Sender	CA	Sender	*Gen*	*Sign*	*KeyGen*	*CA Cert. Vrfy*	*Sender Cert. Vrfy*
Incep.-RSA	1024	***Not Required***				157.1	20.1	21.9	*Not Required*	1.3
	2048					1188.9	64.5	21.4		7.7
	4096					11990.4	314.9	22.3		56.7
	8192					195872.1	1970.7	22.9		437.2
VGG-RSA	1024					153.5	19.9	16.0		1.3
	2048					1186.1	65.1	17.0		7.7
	4096					15130.7	365.7	23.2		65.8
	8192					225352.5	2286.1	28.9		505.9
Res.-RSA	1024					151.0	19.9	6.9		1.3
	2048					1064.3	64.6	7.1		7.7
	4096					15844.0	365.6	9.2		65.9
	8192					245305.9	2290.9	10.2		508.0
RSA*	1024	1159	969	9.6	1.7	129.8	18.1	*Not Required*	1.3	0.3
	2048	1513	1322	48.5	2.4	970.4	50.2		1.2	0.7
	4096	2566	2348	495.0	6.4	13335.4	239.0		1.2	2.4
	8192	4182	3960	5974.7	32.9	194467.5	1289.5		1.6	8.3

** Conventional hashed RSA algorithm (i.e., k_{pub} is 65537)*

*** Public keys are directly estimated from the RF-PubKG; certificate management is NOT required.*

to the sender and receiver before RF transmission. We artificially generate a 12-digit MAC address as a user identity message and evaluate the implemented signature scheme to verify this address. Our focus is on the analysis of time consumption and file size for constructing the PKIs. Specifically, we measure the time required for the Gen, Sign, and Vrfy processes as defined in Algorithm 2. In addition, we measure the time needed for CA certificate verification, the file size for PKI configuration, and the certificate generation time.

Consequently, the proposed RF-based RSA signature is identical to the conventional RSA signature method except for the public key generation process from the RF-PubKGs. For this reason, it was confirmed that the time consumption is similar to that of the conventional RSA algorithm, and it even increased a s the key size increased. This result can be anticipated, given that the method involves a larger key size than general RSA key pairs, which utilize a fixed public key, i.e., k_{pub} is 65537.

The proposed RF-based RSA signature scheme presents an advantage in simplifying the PKI structure. Through the previous discussion, we confirmed that a public key can be uniquely derived from the non-replicable RF features. This means that the PKI, a system for maintaining and managing certificates, can be simplified because the reliability of the public key can be sufficiently secured. As a result of the actual experiment, it is confirmed that Alice's signature could be verified from the public keys estimated from the received RF feature, and a certificate for Alice is not required to verify the public key in this process. We note that one person only needs a few Kbytes and tens of milliseconds, but these amounts can increase exponentially as the number of people managed by PKIs increases.

This evaluation illustrates that the hierarchical model of CAs described in Chapter. II can be sufficiently simplified. The complexity can be minimized, as the structure solely necessitates a RFF model manager responsible for the systematic updates of the RFF models.

D. DISCUSSION

We successfully evaluated the effectiveness and reliability of RF-PubKG and validated the concept of an RF-PubKG based digital signature scheme using the RSA algorithm. This subsection will discuss the impact and future work related to RF-PubKG, along with its drawbacks.

RF-PubKG is a novel RFF process designed to generate trustworthy public keys from non-replicable RF features. It allows for the integrity verification of the public key, based on the device's authenticity at the physical layer. We believe that RF-PubKG can enhance the efficiency of cryptography system structures by being integrated into key verification processes.

As a case study demonstrated in Chapter V-C, we implemented an RF-PubKG based digital signature scheme using the RSA algorithm. This scheme efficiently validates the signature verification directly from the trustworthy public key derived from the RF-PubKG, thereby making certificates redundant and reducing the need for third-party CA management. Consequently, as depicted in Fig. 3, RF-PubKG considerably simplifies the operational complexities and resources required for the PKI entities. We believe this potential application to simplify PKC structures holds promise for a wide range of key-based cryptography.

Future work will focus on addressing the inefficiencies of the RF-PubKG based digital signature scheme in the context of cryptography. As a proof-of-concept, we employed a

cascade structure that combines RF-PubKG and the existing RSA algorithm. Although this approach shows feasibility, it was not optimally efficient from a cryptographic aspect, as demonstrated by the increased time consumption shown in Table 6. These inefficiencies are drawbacks for real-world applications that necessitate further research into more effective cryptographic algorithms for managing the RF-PubKGs.

VI. CONCLUSION

In this research, we have investigated the novel application of RF features in generating trustworthy cryptographic sequences, demonstrating the promising potential of RF features at the physical layer to enhance the efficiency of digital security. We proposed RF-PubKG, which utilizes a key generation layer within the RFF model to effectively map analog RF features to digital cryptographic key sequences. This work establishes a novel paradigm for public key generation.

We evaluated the effectiveness of RF-PubKG. We achieved key estimation accuracy of over 99% for various cryptographic key lengths, with a generation time of only 10.8ms. In AWGN channels with an SNR level over 20 dB, these results maintained a 97.2% accuracy along with a 5.6% FER, which decreased below 1% as channel conditions improved.

We corroborated the reliability of the RF-PubKG by validating the consistency and clustering centrality of the public key sets when compared to the testing dataset. We confirmed the independence among public keys by measuring correlation values lower than 0.24. Notably, the ability to generate distinct key sets with updates to the RFF model was demonstrated by correlation values lower than 0.04, emphasizing the dynamic and adaptable nature of the RF-PubKG scheme.

As a proof-of-concept, we have validated the RF-PubKG-based digital signature scheme using an RSA algorithm. This scheme enhances PKI efficiency by generating reliable public keys directly from unique RF features, thus avoiding the complexities of third-party CA management. These results validate the signature verification directly from the RF-derived public keys, making certificates redundant. Such simplification could reduce the operational complexities and resource demands for PKIs, enhancing the efficiency of digital signature applications by simplifying the PKI entities. This process of verification could become less complex and resource-intensive when managing a large number of identities.

This research is a pioneering exploration into utilizing RF features as cryptographic sequences, thereby substantiating the cryptographic viability of the proposed method. Research findings not only evaluate the efficiency and reliability of the RF-PubKG but also its applicability to real-world cryptographic scenarios.

As a direction for future research, we plan to further improve our research findings by integrating ECC to improve FER rates and expanding the application of PKC to simplify complex hierarchical systems of cryptography. We will continue to pave the way for more secure and efficient cryptographic solutions derived from the potential of RF signal features.

REFERENCES

[1] J. Zhang, C.-H. Chang, C. Gu, and L. Hanzo, "Radio frequency fingerprints vs. physical unclonable functions–are they twins, competitors, or allies?" *IEEE Netw.*, vol. 36, no. 6, pp. 68–75, Nov. 2022.

[2] *IEEE Standard for Local and Metropolitan Area Networks-Port-Based Network Access Control*, IEEE Standard 802.1X-2020, 2020.

[3] *IEEE Standard for Local and Metropolitan Area Networks-Media Access Control (MAC) Security*, IEEE 802.1 Working Group, 2018.

[4] K. Seo and S. Kent, *Security Architecture for the Internet Protocol*, document RFC 4301, Internet Engineering Task Force, Dec. 2005.

[5] S. Khan, F. Luo, Z. Zhang, M. A. Rahim, M. Ahmad, and K. Wu, "Survey on issues and recent advances in vehicular public-key infrastructure (VPKI)," *IEEE Commun. Surveys Tuts.*, vol. 24, no. 3, pp. 1574–1601, 3rd Quart., 2022.

[6] I. O. Kennedy, P. Scanlon, F. J. Mullany, M. M. Buddhikot, K. E. Nolan, and T. W. Rondeau, "Radio transmitter fingerprinting: A steady state frequency domain approach," in *Proc. IEEE 68th Veh. Technol. Conf.*, Calgary, AB, Canada, Sep. 2008, pp. 1–5.

[7] A. M. Ali, E. Uzundurukan, and A. Kara, "Assessment of features and classifiers for Bluetooth RF fingerprinting," *IEEE Access*, vol. 7, pp. 50524–50535, 2019.

[8] J. Yu, A. Hu, G. Li, and L. Peng, "A robust RF fingerprinting approach using multisampling convolutional neural network," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 6786–6799, Aug. 2019.

[9] J. Kang, Y. Shin, H. Lee, J. Park, and H. Lee, "Radio frequency fingerprinting for frequency hopping emitter identification," *Appl. Sci.*, vol. 11, no. 22, p. 10812, Nov. 2021.

[10] A. Jagannath and J. Jagannath, "Embedding-assisted attentional deep learning for real-world RF fingerprinting of bluetooth," *IEEE Trans. Cognit. Commun. Netw.*, vol. 9, no. 4, pp. 940–949, Apr. 2023.

[11] Y. Zeng, Y. Gong, J. Liu, S. Lin, Z. Han, R. Cao, K. Huang, and K. B. Letaief, "Multi-channel attentive feature fusion for radio frequency fingerprinting," *IEEE Trans. Wireless Commun.*, early access, Sep. 25, 2023, doi: 10.1109/TWC.2023.3316286.

[12] N. Soltanieh, Y. Norouzi, Y. Yang, and N. C. Karmakar, "A review of radio frequency fingerprinting techniques," *IEEE J. Radio Freq. Identificat.*, vol. 4, no. 3, pp. 222–233, Sep. 2020.

[13] A. Jagannath, J. Jagannath, and P. S. P. V. Kumar, "A comprehensive survey on radio frequency (RF) fingerprinting: Traditional approaches, deep learning, and open challenges," *Comput. Netw.*, vol. 219, Dec. 2022, Art. no. 109455.

[14] 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. Comput. Commun. (INFOCOM)*, Apr. 2019, pp. 370–378.

[15] N. Soltani, G. Reus-Muns, B. Salehi, J. Dy, S. Ioannidis, and K. Chowdhury, "RF fingerprinting unmanned aerial vehicles with non-standard transmitter waveforms," *IEEE Trans. Veh. Technol.*, vol. 69, no. 12, pp. 15518–15531, Dec. 2020.

[16] O. M. Gul, M. Kulhandjian, B. Kantarci, A. Touazi, C. Ellement, and C. D'amours, "Secure industrial IoT systems via RF fingerprinting under impaired channels with interference and noise," *IEEE Access*, vol. 11, pp. 26289–26307, 2023.

[17] J. Zhang, S. Rajendran, Z. Sun, R. Woods, and L. Hanzo, "Physical layer security for the Internet of Things: Authentication and key generation," *IEEE Wireless Commun.*, vol. 26, no. 5, pp. 92–98, Oct. 2019.

[18] B. Danev, H. Luecken, S. Capkun, and K. El Defrawy, "Attacks on physical-layer identification," in *Proc. 3rd ACM Conf. Wireless Netw. Secur.*, Mar. 2010, pp. 89–98.

[19] K. Merchant and B. Nousain, "Enhanced RF fingerprinting for IoT devices with recurrent neural networks," in *Proc. IEEE Mil. Commun. Conf. (MILCOM)*, Nov. 2019, pp. 590–597.

[20] F. Zhuo, Y. Huang, and J. Chen, "Radio frequency fingerprint extraction of radio emitter based on I/Q imbalance," in *Proc. Int. Congr. Inf. Commun. Technol. (ICICT)*, 2017, pp. 472–477.

[21] H. J. Patel, M. A. Temple, and R. O. Baldwin, "Improving ZigBee device network authentication using ensemble decision tree classifiers with radio frequency distinct native attribute fingerprinting," *IEEE Trans. Rel.*, vol. 64, no. 1, pp. 221–233, Mar. 2015.

[22] H. Jafari, O. Omotere, D. Adesina, H.-H. Wu, and L. Qian, "IoT devices fingerprinting using deep learning," in *Proc. IEEE Mil. Commun. Conf. (MILCOM)*, Los Angeles, CA, USA, Oct. 2018, pp. 1–9.
[23] L. Peng, J. Zhang, M. Liu, and A. Hu, "Deep learning based RF fingerprint identification using differential constellation trace figure," *IEEE Trans. Veh. Technol.*, vol. 69, no. 1, pp. 1091–1095, Jan. 2020.
[24] L. Zong, C. Xu, and H. Yuan, "A RF fingerprint recognition method based on deeply convolutional neural network," in *Proc. IEEE 5th Inf. Technol. Mechatronics Eng. Conf. (ITOEC)*, Jun. 2020, pp. 1778–1781.
[25] 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. Comput. Commun. (INFOCOM)*, Jul. 2020, pp. 646–655.
[26] D. Roy, T. Mukherjee, M. Chatterjee, and E. Pasiliao, "Detection of rogue RF transmitters using generative adversarial nets," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Apr. 2019, pp. 1–7.
[27] K. Merchant and B. Nousain, "Securing IoT RF fingerprinting systems with generative adversarial networks," in *Proc. IEEE Mil. Commun. Conf. (MILCOM)*, Norfolk, VA, USA, Nov. 2019, pp. 584–589.
[28] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.
[29] J. Katz and Y. Lindell, *Introduction to Modern Cryptography*. Boca Raton, FL, USA: Chapman & Hall, 2007.
[30] D. R. Stinson, *Cryptography: Theory and Practice*, 3rd ed. Boca Raton, FL, USA: CRC Press, Nov. 2005.
[31] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 818–833.
[32] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *Proc. 27th Int. Conf. Neural Inf. Process. Syst.*, 2014, pp. 3320–3328.
[33] M. Norouzi et al., "Hamming distance metric learning," in *Proc. Adv. Neural Inf. Process. Syst.*, Dec. 2012, pp. 1061–1069.
[34] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Commun. ACM*, vol. 26, no. 1, pp. 96–99, Jan. 1983.
[35] *Secure Hash Standard*, document FIPS PUB 180-1, National Institute of Standards and Technology (NIST), U.S. Department of Commerce, Apr. 1995.
[36] *Electromagnetic Compatibility and Radio Spectrum Matters (ERM); Digital Mobile Radio (DMR) Systems; Part 1: DMR Air Interface (AI) Protocol*, Standard ETSI TS 102 361-1, European Telecommunications Standards Institute, 2016.
[37] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Represent.*, 2015, pp. 1–14.
[38] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 770–778.
[39] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, "Inception-v4, inception-ResNet and the impact of residual connections on learning," in *Proc. 31st AAAI Conf. Artif. Intell.*, San Francisco, CA, USA, Feb. 2017, pp. 4278–4284.
[40] Y. Pan, S. Yang, H. Peng, T. Li, and W. Wang, "Specific emitter identification based on deep residual networks," *IEEE Access*, vol. 7, pp. 54425–54434, 2019.
[41] K. Merchant, S. Revay, G. Stantchev, and B. Nousain, "Deep learning for RF device fingerprinting in cognitive communication networks," *IEEE J. Sel. Topics Signal Process.*, vol. 12, no. 1, pp. 160–167, Feb. 2018.
[42] M. A. Ganaie, M. Hu, A. K. Malik, M. Tanveer, and P. N. Suganthan, "Ensemble deep learning: A review," *Eng. Appl. Artif. Intell.*, vol. 115, 2022, Art. no. 105151. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S095219762200269X
[43] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, Nov. 2008.
[44] *PyCrypotdome Documentation*. Accessed: Aug. 25, 2023. [Online]. Available: https://pycryptodome.readthedocs.io/en/latest/
[45] *pyOpenSSL Documentation*. Accessed: Aug. 25, 2023. [Online]. Available: https://www.pyopenssl.org/en/latest/index.html
[46] *Digital Signature Standard (DSS)*, document NIST FIPS PUB 186-4, National Institute of Standards and Technology (NIST), U.S. Department of Commerce, 2013.

JUSUNG KANG received the B.S. degree in electrical engineering from Ajou University, Suwon-si, Gyeonggi-do, South Korea, in 2012. Currently, he is pursuing the integrated M.S. and Ph.D. degree with the School of Electrical Engineering and Computer Science, Gwangju Institute of Science and Technology (GIST), Gwangju, South Korea. His research interests include the application of AI and cryptography to radio frequency signal processing. Specific areas of focus within these domains include radio signal classification, outlier detection, incremental learning, reinforcement learning, and public key cryptography for zero-knowledge proof.

YOUNG-SIK KIM (Member, IEEE) received the B.S., M.S., and Ph.D. degrees in electrical engineering and computer science from Seoul National University, in 2001, 2003, and 2007, respectively. He joined the Semiconductor Division, Samsung Electronics, where he performed research and development of security hardware IPs for various embedded systems, including modular exponentiation hardware accelerator called Tornado 2MX2 for RSA and elliptic curve cryptography in smart card products and mobile application processors of Samsung Electronics, until 2010. He was a Professor with Chosun University, from September 2010 to August 2023. He is currently a Professor with the Daegu Gyeongbuk Institute of Science and Technology (DGIST), Daegu, South Korea. He is also a Submitter for two candidate algorithms (McNie and pqsigRM) in the first round for the NIST Post Quantum Cryptography Standardization. His research interests include post-quantum cryptography, the IoT security, physical layer security, data hiding, channel coding, and signal design. He was selected as one of 2025's 100 Best Technology Leaders (for CryptoSystems) by the National Academy of Engineering of Korea.

HEUNG-NO LEE (Senior Member, IEEE) received the B.S., M.S., and Ph.D. degrees in electrical engineering from the University of California at Los Angeles, CA, USA, in 1993, 1994, and 1999, respectively. He was a Research Staff Member with the HRL Laboratories, LLC, Malibu, CA, USA, from 1999 to 2002. From 2002 to 2008, he was an Assistant Professor with the University of Pittsburgh, PA, USA. In 2009, he joined the School of Electrical Engineering and Computer Science, Gwangju Institute of Science and Technology, Gwangju, South Korea. He is currently with the Gwangju Institute of Science and Technology. His research interests include information theory, signal processing theory, blockchain, communications/networking theory, and their application to wireless communications and networking, compressive sensing, future internet, and brain–computer interface. He has received several prestigious national awards, including the Top 100 National Research and Development Award, in 2012, the Top 50 Achievements of Fundamental Research Award, in 2013, and the Science/Engineer of the Month, in January 2014.