Provably Secure Public-Key Steganography Based on Elliptic Curve Cryptography

Xin Zhang, Kejiang Chen, *Member, IEEE*, Jinyang Ding, *Member, IEEE*, Yuqi Yang, Weiming Zhang, and Nenghai Yu

***Abstract*— Steganography is the technique of hiding secret messages within seemingly harmless covers to elude examination by censors. Despite having been proposed several decades ago, provably secure steganography has not gained popularity among researchers due to its rigorous data requirements. Recent advancements in generative models have enabled these researchers to provide explicit data distributions, which has contributed to the development of provably secure steganography methods. However, these methods depend on the assumption of a preshared key. In practical settings, these methods face various challenges, including key agreement, key updating, and user expansion. Although public-key steganography provides a viable solution, existing public-key steganography approaches are burdened with inefficiency and complex implementation in practical scenarios. In this paper, we proposes a practical public-key steganography method based on elliptic curve cryptography and a generative model. This method is the first comprehensive and practical approach to public-key steganography and steganographic key exchange. Additionally, we provide a specific instance to illustrate the proposed method. The security of the proposed construction is also proven based on computational complexity theory. Further experiments have demonstrated the security and efficiency of the proposed method.**

***Index Terms*— Public-key steganography, generative model, elliptic curve cryptography, provable security.**

I. Introduction

STEGANOGRAPHY [1], [2], [3] is a covert communication technique that hides secret information within seemingly innocent objects, such as text, images, audio, and videos. This technology has gained increased attention in recent years due to the rise in internet censorship [4], [5], [6], as it not only protects the content of secret information but also conceals the fact that covert communication is taking place.

In steganographic communication, the steganographer embeds secret information into the widely distributed covers to obtain the stegos and strives for the covers and the stegos to be indistinguishable. The steganalyzer [7], conversely, is dedicated to finding subtle differences in features between the covers and the stegos. The corresponding technology is called steganalysis [8].

In digital steganography, past methods such as least significant bit (LSB) replacement [9] and exploiting modification direction (EMD) [10] were easily detected by artificial statistical features [11]. To address this, researchers proposed the minimum distortion model [12], focusing on designing distortion functions [13] and steganographic codes [14]. Advances include machine learning-based distortion [15], [16], [17], [18], yet these methods remain vulnerable to deep learning-based detection [19], [20], [21]. To improve security, it is crucial to develop provably secure steganographic communication systems beyond empirical frameworks.

A. Classical Provably Secure Steganography

There have been two categories of provable security in steganography thus far: information-theoretic security and computational security. The information-theoretic security model was proposed by Cachin [22], who first modeled steganographic security by $D_{\mathrm{KL}}(P_{\mathrm{c}} \| P_{\mathrm{s}})$, which is the Kullback-Leibler divergence between the cover distribution P_{c} and the stego distribution P_{s}. Hopper et al. [23] proposed a steganographic model based on computational complexity theory and constructed a provably secure steganographic method based on the perfect sampler hypothesis and the rejection sampling method.

The perfect sampler assumption in provable security steganography requires precise sampling of a particular cover distribution. Initially, email and camera were considered text and image samplers respectively, but they did not meet that requirement. Therefore, provably secure steganography remained a theoretical concept at that time.

B. Generative Provably Secure Steganography

The development of deep learning and generative models [24], [25], [26] has significantly enhanced the ability to learn probability distributions from samples and perform accurate sampling from those distributions. This progress has, in turn, greatly advanced the field of provably secure steganography.

Manuscript received 17 May 2023; revised 20 September 2023 and 19 December 2023; accepted 29 January 2024. Date of publication 1 February 2024; date of current version 9 February 2024. This work was supported in part by the Natural Science Foundation of China under Grant U2336206, Grant 62102386, Grant 62072421, Grant 62372423, and Grant 62121002. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Roberto Caldelli. *(Corresponding author: Kejiang Chen.)*

The authors are with the CAS Key Laboratory of Electro-Magnetic Space Information, University of Science and Technology of China, Hefei 230026, China, and also with the Anhui Province Key Laboratory of Digital Security, Hefei 230026, China (e-mail: chenkj@mail.ustc.edu.cn).

Digital Object Identifier 10.1109/TIFS.2024.3361219

Yang et al. [27] were the first to propose the idea of implementing provably secure steganography by using an autoregressive generative model and arithmetic coding (AC). Chen et al. [28] and Ziegler et al. [29] extended this method to text-to-speech and text-generation tasks, respectively. Zhang et al. [30] introduced a steganography method based on adaptive dynamic grouping (ADG), which can achieve provable security if and only if the grouping is perfectly balanced. Kaptchuk et al. [31] proposed Meteor, a novel approach that addresses the randomness reuse problem found in AC-based methods. Ding et al. [32] presented a more efficient method based on distribution copies called Discop, which outperforms previous attempts in the field. Recently, de Witt et al. [33] demonstrated that achieving maximal transmission efficiency among perfect security procedures is equivalent to solving a minimum entropy coupling (MEC) problem and introduced a method based on an iterative MEC (iMEC) procedure.

C. From Symmetric Steganography to Public-Key Steganography

In the communication scenario, a *session* is a temporary and interactive exchange of information between two or more devices, users, or applications over a communication channel, such as the Internet. Sessions involve a series of requests and responses and are created and maintained to enable continuous communication between the involved parties.

Session security in steganography typically encompasses various aspects, including secure session establishment, session concealment, message integrity within the session, and identity authentication within the session. Secure session establishment requires initiating the session with the negotiation of a shared key, while session concealment demands that session content and channel distributions be indistinguishable to conceal covert communication as normal communication processes. Message integrity requires verification that communication content remains unaltered, and identity authentication requires confirming the identities of the communicators.

The symmetric steganography method such as AC [27], Meteor [31] and Discop [32], are primarily limited to session concealment. Normally they assume that communication users have already shared a symmetric key, which remains unknown to any potential attacker. The sender encrypts the plaintext message by using this key and then embeds the encrypted message (i.e., ciphertext) within the media. Although these approaches have demonstrated promise, they have not proposed solutions for other aspects of session security, inherently restricting their applicability in general communication scenarios.

The concept of public-key steganography, which can effectively address these aspects of session security, was initially introduced by von Ahn and Hopper as early as 2004 [34]. Their model introduced a pivotal shift by substituting the cryptographic algorithms traditionally used in symmetric key provable security steganography methods with public-key ciphers featuring pseudo-random output algorithms. Subsequent research endeavors further expanded the scope by incorporating considerations for active attack scenarios [35], [36].

However, despite the theoretical promises of these public-key steganography constructions, they have not gained widespread adoption in practice. One of the key challenges lies in the inability to achieve perfect sampling, which has hindered their practical implementation. The samplers proposed by Hopper et al. [23], [35], such as those tailored for email [23] and document [34] use cases, prove to be impractical and inefficient. These samplers fail to deliver the precise sampling of the carriers, thereby posing a significant hurdle in the practical application of this technique.

D. Our Method

Inspired by provably secure symmetric generative steganography constructs and to make public-key steganography truly practical, we present a comprehensive public-key steganography scheme that leverages elliptic curve cryptography and a generative model. More specifically, we present an indistinguishable from random bits under the chosen plaintext attack (IND$-CPA) secure public-key encryption scheme that employs two hybrid encryption schemes. The first ciphertext is generated by using a pseudorandom elliptic curve point encoding technique called Elligator 2 [37], which transforms points on the curve into a binary string indistinguishable from a uniformly random string. To meet the Decisional Diffie-Hellman (DDH) assumption [38] on elliptic curves, we multiply the selected point by a constant factor and hash the result to obtain a key for the second ciphertext. The second ciphertext encrypts the message by using an IND$-CPA secure symmetric encryption scheme, ensuring that the overall encryption scheme is IND$-CPA secure.

By using this public-key encryption system, we build a public-key steganography system based on reversible mapping and sampling, which reversibly maps binary strings to mutually exclusive and measure-equivalent subdistributions of generative data. Afterward, we construct an instance of a public-key steganography system based on a text-to-speech model, WaveGlow [28]. By introducing reversible mapping on zero-mean spherical Gaussian sampling, we can generate highly naturalistic audio that can hide a message and thus put our construction into practice. We deploy the whole system on Curve25519 [39].

Additionally, we propose a one-round asynchronous algorithm for steganographic key exchange, which is beneficial in situations where revealing public keys poses challenges.

We also provide comprehensive proof that the proposed public-key steganography construction is resistant to chosen hiddentext attacks (CHA) [34] under complexity theory. The security of the proposed scheme is also verified through National Institute of Standards and Technology (NIST) pseudorandomness tests [40] and steganalysis experiments [41], [42], [43].

E. Contributions

The main contributions of this paper can be summarized as follows:

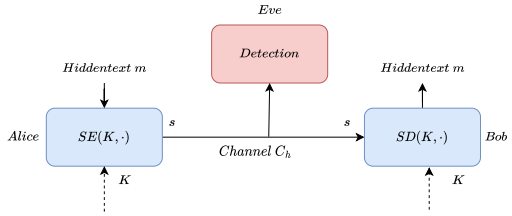


Fig. 1. Diagram of the symmetric steganography system.

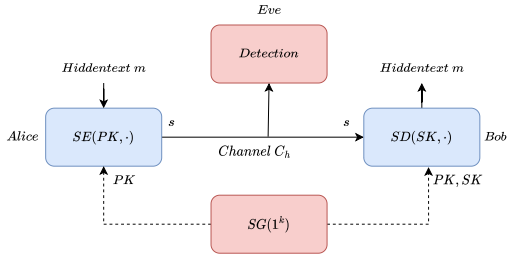


Fig. 2. Diagram of the public-key steganography system.

- **Session security.** To comprehensively address all facets of session security, we propose an efficient public-key steganography method with favorable computational complexity and embedding efficiency. Our proposed method leverages elliptic curve cryptography and utilizes a generative model to create a practical public-key steganography approach. To the best of our knowledge, this is the first comprehensive and practical method for public-key steganography.
- **Comprehensive proof.** For the public-key steganography construction we designed in the paper under the chosen hiddentext attack, we provide comprehensive proof by using computational complexity theory.
- **Steganographic key exchange construction.** We propose a steganographic key exchange based on the elliptic-curve Diffie-Hellman (ECDH) and the generative model, which can be used for covert key exchange in situations where revealing public keys poses challenges.

II. Related Work

A. Symmetric Steganography System

The classical model of steganography, presented by Simmons [7], describes the prisoners' problem in which Alice and Bob, as prisoners, plot their escape, as shown in Fig.1. Their sole means of communication is a public channel under the watchful eye of warden Eve. To deceive Eve and to maintain the appearance of ordinary conversation, Alice and Bob must skillfully embed their secret information within the seemingly innocuous text by using a pre-shared key. Then, to accurately decode the hidden data, Alice and Bob both use the same key to extract the secret information from the text.

The classical model is symmetric, as shown in Fig.1. The encoding algorithm takes a shared key with a message (known as hiddentext) to produce stego, while the decoding algorithm takes stego and the same key to extract the message. The security goal is to ensure that stego and cover samples from the channel C_h are indistinguishable.

Similar to cryptography, steganography must follow Kerckhoffs's principle [44], which implies that any information except the key must be presumed to be disclosed to Eve.

B. Public-Key Steganography System

The symmetric steganography system mainly focuses on session security, which emphasizes the importance of sharing a secret key between communicating parties to ensure the confidentiality and integrity of the exchanged data. However, its deficiencies become evident when multiple users persistently seek to participate in concealed communication on social networks.

- **No key negotiation before the session begins.**
- **Poor scalability.** If a new user seeks to join a covert communication network, they are required to establish key exchanges with all individuals they intend to communicate with.
- **Hardness for maintaining and key refreshing.** For a group of n users, the total number of keys that need to be maintained is $O(n^2)$, increasing at a square rate. It is difficult to guarantee forward security during key updates.
- **No efficient identity authentication mechanism.** The absence of an effective identity authentication mechanism in symmetric steganography leaves the system vulnerable to potential attacks involving identity forgery.

Inspired by public-key encryption, which effectively resolves these issues, von Ahn and Hopper [34] proposed public-key steganography.

As shown in Fig.2, a public-key steganography system is a triple of probabilistic algorithms $SS = (SG, SE, SD)$. $SG(1^k)$ takes a random bitstream as input and generates a key pair (PK, SK). $SE(PK, m, h)$ takes the public key as input, the hidden message m, and the distribution of the history-based channel C_h. It then outputs the stegotext s sampled from C_h. In this context, the channel refers to the cover medium used for concealing the hidden message. $SD(SK, s, h)$ takes the private key, stegotext, and history, and outputs hiddentext m:

$$\forall\, m, h,\ \ (PK, SK) \leftarrow SG(1^k).$$

SE, SD should be self-consistent, that is:

$$\Pr[SD(SK, SE(PK, m, h), h) = m] \geq 1 - \epsilon(k) \qquad (1)$$

which suggests that the decoding error probability can be regarded as negligible.

Compared to symmetric steganography shown in Fig.1, public-key steganography simplifies key distribution, streamlines key management, and enables the use of digital signatures.

The existing constructions of public-key steganography, as seen in works by Hopper et al. [23], [35], rely on RSA [45] with the probabilistic bias removal method (PBRM) and Elgamal [46]. The methods based on RSA suffer from poor pseudo-randomness and those based on Elgamal cannot compete with elliptic curve-based encryption algorithms [47] regarding both encryption speed and key length. To enhance computational efficiency, and encoding efficiency, and diversify the solutions while maintaining the same level of security,

we propose public-key steganography based on elliptic curve cryptography.

C. Provably Secure Steganography

The concept of provable security in steganography can be divided into two categories: information-theoretic security and computational security.

1) Information-Theoretic Security: Cachin [22] first introduced the information-theoretic definition of steganography security based on relative entropy (a.k.a. Kullback-Leibler divergence) between the cover distribution P_c and the stego distribution P_s,

$$D_{\mathrm{KL}}(P_c \| P_s) = \sum_{\mathbf{x}\in\mathcal{C}} P_c(\mathbf{x}) \log \frac{P_c(\mathbf{x})}{P_s(\mathbf{x})}, \tag{2}$$

When $D_{\mathrm{KL}}(P_c \| P_s) = 0$, the stegosystem is considered to be *perfectly secure*. In this case, the distributions P_c and P_s are the same.

2) Computational Security: In light of the fact that information-theoretic security, as proposed by Cachin [22], primarily provides a theoretical upper bound, Hopper et. al. [23], [48] introduced computational complexity theory to distinguish cover and stego distributions by allowing attackers to engage in a probabilistic game, thereby defining the chosen plaintext attack (CPA) and the computational security against it. We provide detailed definitions in Section III.

3) Provably Secure Steganographic Construction: Since the definition of computational security was provided by Hopper et.al. [23], many provably secure steganographic constructions have arisen [23], [36], [49], encompassing both public-key and symmetric steganography. One common example is rejection sampling construction, which is based on the perfect sampler assumption.

Define a channel as a distribution with timestamp: $C = ((c_1, t_1), (c_2, t_2), \ldots)$, where $\forall i > 0 : t_{i+1} > t_i$. Anyone communicating on a channel can be regarded as implicitly sampling from the channel conditioned on history h, defined as $c \leftarrow C_h$.

Given an oracle $\mathcal{O}^{\mathcal{C}}$ over distribution C, A function $f : C \to R$ is called $\epsilon - biased$ if $|Pr_{x\leftarrow C}[f(x) = 0] - 1/|R|| < \epsilon$. We say f is perfectly unbiased if $\epsilon = 0$. Given hiddentext m, the rejection sampling is defined as follows:

$$\text{sample } x \text{ from } \mathcal{O}^{\mathcal{C}} \text{ until } f(x) = m.$$

Supposing m is computationally indistinguishable from a uniform random string, the result of rejection sampling based on channel C is computationally indistinguishable from the channel distribution.

Hopper et al. [23], [34] assumed that the oracle over the channel $\mathcal{O}^{C}$ is ideal, performing precise sampling on a specific cover distribution independently and in parallel while allowing state backtracking. Initially, email and camera were deemed text and image samplers, respectively, but neither met the perfect sampling requirements. As such, symmetric and public-key steganography constructions remained theoretical concepts.

D. Generative Provably Secure Symmetric Steganography

Recently, we have witnessed the rapid development of deep learning and generative models such as generative adversarial networks (GANs) [50], variational autoencoders (VAEs) [51] and flow-based models [26].

Implicit generative models, such as GANs, VAEs, and flow-based models, can generate new data samples by implicitly modeling the underlying probability distribution of the data. Specifically, they map from a noise distribution to the data distribution by avoiding explicit modeling of the latter.

$$x = f_\theta(z),\ z \sim p(z),$$

x represents the object sampled from the generative model, while z designates the latent variable that follows accessible probability distribution.

The remaining concern is whether the generated data are suitable for steganography. However, this challenge has become less significant as the use of artificial intelligence-generated content (AIGC) models, such as GPT-4 [52] and Midjourney [53], are increasingly prevalent. The generated data are widely spread in various real-world scenarios, including the generation of news [54], literature [55], and music [56]. The widespread adoption of AIGC makes it feasible for steganography to disguise itself as such without raising suspicion.

In recent years, significant strides have been made in advancing symmetric steganography, largely attributed to the progress in generative models. The central concept revolves around the mapping of a concealed message to a latent variable, designated as z. When it comes to decoding, this latent variable is retrieved through the probability distribution of a shared model, streamlining the process of message extraction.

This form of steganography effectively addresses the challenge of perfect samplers mentioned earlier. In the context of Adaptive Information Gathering Channels (AIGC), the channel's distribution represents the normal sampling process of the generative model, while generative steganography introduces a specialized sampling process driven by the secret message. The security of steganography lies in distinguishing between these two sampling methods.

Several efforts have been made to use generative models with provably secure Symmetric Steganography, including AC [28], [29], ADG [30], Meteor [31], MEC [33] and Discop [32]. Inspired by provably secure symmetric steganography constructs, our study attempts to provide a practical and complete design of provably secure public-key steganography using generative models.

III. Definition

A. Negligible Function

A function $f : \mathcal{N} \to [0, 1]$ is negligible if for any polynomial $poly(\cdot)$, $\exists N \in \mathcal{N}$, $\forall n > N$, $f(n) < 1/poly(n)$.

B. Elliptic Curve Group and Computations

1) Elliptic Curve and Points on the Curve: Let F_p denote the finite field modulo prime p and E denote an elliptic curve

over F_p described in Weierstrass form [57]:

$$E_{A,B} : y^2 = x^3 + Ax + B, \quad A, B \in F_p,$$

With the discriminant, $\Delta = 4A^3 + 27B^2 \neq 0$. The set of rational points in $E_{A,B}$ over F_p is given by

$$E_{A,B}(F_p) = \{(x, y) \in F_p^2 : E_{A,B}(x, y) = 0\} \bigcup O,$$

where O is the point at infinity.

Define $\#E_{A,B}(F_p)$ or $\|E_{A,B}(F_p)\|$ as the cardinality of the set of points on the elliptic curve $E_{A,B}(F_p)$ defined over the finite field F_p. In other words, it represents the number of points on the curve.

2) Addition: Let P and Q be points on an elliptic curve E, with coordinates (x_P, y_P) and (x_Q, y_Q), respectively. The addition $P + Q$ is defined as follows:

- If P and Q are the same point, then $P + Q = 2P$.
- If either P or Q is the point at infinity, then $P + Q$ is defined to be the other point.
- Otherwise, draw the line passing through P and Q, and let it intersect the curve at a third point R. Then, $P + Q$ is defined as the reflection of R across the x-axis.

This definition of point addition satisfies the group axioms, making the set of points on an elliptic curve into an abelian group $(E_{A,B}(F_p), +)$.

3) Scalar Multiplication: $(E_{A,B}(F_p), +)$ form an Abelian group. Thus, we define scalar multiplication $(\cdot)$ over $E_{A,B}(F_p)$ as follows:

$$k \cdot P = P + P + P + \ldots + P \ (k \text{ times}),$$
$$k \in \mathcal{N}, \ P \in E_{A,B}(F_p).$$

C. Decisional Diffie-Hellman Assumption in Elliptic Curve Group (ECDDH)

Let $G \triangleq E_{A,B}(F_p)$ be the group of an elliptic curve points such that a sufficiently large prime Q divides $\#G$. Let k denote the binary length of P. Let $g \in G$ have order Q.

Let $\mathcal{A}$ be a probabilistic polynomial time machine (PPTM) that takes as input three elements of G and outputs a single bit. The ECDDH advantage of $\mathcal{A}$ over (g, G, Q) is defined as:

$$\mathbf{Adv}^{\text{ddh}}_{g,G,Q}(\mathcal{A}) = \left| \begin{array}{l} \Pr\limits_{a,b,r}\left[\mathcal{A}(a \cdot g, b \cdot g, a \cdot b \cdot g) = 1\right] \\ - \Pr\limits_{a,b,c,r}\left[\mathcal{A}(a \cdot g, b \cdot g, c \cdot g) = 1\right] \end{array} \right|, \quad (3)$$

where r, a, b, c are chosen uniformly at random from Z_Q.

The ECDDH *insecurity* of (g, G, Q) is defined as

$$\mathbf{InSec}^{\text{ddh}}_{g,G,Q}(t) = \max_{\mathcal{A} \in \mathcal{A}(t)} \{\mathbf{Adv}^{\text{ddh}}_{g,G,Q}(\mathcal{A})\}, \quad (4)$$

where the maximum is taken over all $\mathcal{A}$ running in time t.

The decisional Diffie-Hellman assumption in the Elliptic curve group is a computational hardness assumption requiring that $\mathbf{InSec}^{\text{ddh}}_{g,G,Q}(t)$ is negligible in k, where $t = poly(k)$. In other words, for any sufficiently small $\epsilon(k) = \frac{1}{poly(k)}$, there is no PPTM $\mathcal{A}$ running in time t that achieves $\mathbf{Adv}^{\text{ddh}}_{g,G,Q}(\mathcal{A}) \leq \epsilon(k)$.

This assumption forms the basis for the security of many elliptic curve-based cryptographic constructions, such as elliptic curve Diffie-Hellman (ECDH) and elliptic curve-based ElGamal encryption [47], which rely on the notion that computing discrete logarithms on elliptic curves is a difficult problem. If this assumption proves false, the security of these cryptographic constructions may be compromised.

It is widely believed that this assumption holds for a prime-order elliptic curve E over the field $GF(p)$, where p is prime. This is known as the ECDDH assumption.

D. Chosen Hiddentext Attack

We will provide a detailed introduction to our threat model in this subsection.

Threat Model. Consider a public-key steganography system $SS = (SG, SE, SD)$ and an attacker $\mathcal{A}$. $\mathcal{A}$ play a game named chosen hiddentext attack (CHA) described as follows:

- **Key generation stage.** $(PK, SK) \leftarrow SG\left(1^k\right)$.
- **Learning stage.** $\mathcal{A}$ sends hiddentext $m_{\mathcal{A}}$ and history $h_{\mathcal{A}}$ and gets return $SE(PK, m_{\mathcal{A}}, h)$. $\mathcal{A}$ can perform this stage multiple times.
- **Challenge stage.** $\mathcal{A}$ sends hiddentext $m \in \mathcal{A} \setminus \{m_{\mathcal{A}}\}$ to an oracle, which will flip a coin $b \in \{0, 1\}$. If $b = 0$, $\mathcal{A}$ obtains $s = SE(PK, m, h)$; if $b = 1$, $\mathcal{A}$ obtains $c \leftarrow C_h$.
- **Guess stage.** $\mathcal{A}$ outputs a bit b' as its "guess" to determine whether it has received a stegotext or a covertext.

Define the Chosen Hiddentext Attack (CHA) [23] advantage of $\mathcal{A}$ against SS over channel C by:

$$\mathbf{Adv}^{\text{cha}}_{SS,C}(\mathcal{A}, k) = \left| \begin{array}{l} \Pr\limits_{PK}\left[\mathcal{A}(PK, s) = 1\right] \\ - \Pr\limits_{PK}\left[\mathcal{A}(PK, c) = 1\right] \end{array} \right|. \quad (5)$$

Define the *insecurity* of SS over channel C by

$$\mathbf{InSec}^{cha}_{SS,C}(t, l, k) = \max_{\mathcal{A} \in \mathcal{A}_{(t,l)}} \{\mathbf{Adv}^{cha}_{SS,C}(\mathcal{A}, k)\}, \quad (6)$$

where $\mathcal{A}_{(t,l)}$ is the set of all adversaries that send at most $l(k)$ bits and run in time $t(k)$. $l(k)$ and $t(k)$ are polynomials of k. SS is secure against CHA if $\mathbf{InSec}^{cha}_{SS,C}(t, l, k)$ is negligible in k, i.e., no probabilistic polynomial time (PPT) adversary can distinguish s and c with nonnegligible probability.

Remark. In the Chosen Hiddentext Attack scenario, we assume that an adversarial attacker, denoted as $\mathcal{A}$, has taken control of an identical steganographic encoder during the learning stage. $\mathcal{A}$ can interact with this encoder and input the hiddentext $m_{\mathcal{A}}$ multiple times, generating stegos that conceal $m_{\mathcal{A}}$ under any given history $h_{\mathcal{A}}$, i.e., $SE(PK, m_{\mathcal{A}}, h)$. Then, during the Challenge stage, $\mathcal{A}$ is presented with either a stego $s = SE(PK, m, h)$ that may have been generated from some hiddentext m using the same encoder, or it could be a cover $c \leftarrow C_h$ randomly sampled from the channel. With the knowledge acquired, $\mathcal{A}$ needs to guess at the Guess stage with a probability significantly greater than 0.5, determining whether it is a cover or a stego, thus achieving a successful attack.

This threat model is a passive attack and is conceptually similar to the cryptographic CPA (Chosen Plaintext Attack) which we will describe below, except that CPA focuses on guessing additional information related to the key, while CHA requires guessing the presence or absence of secret

information. CHA encompasses scenarios where the steganographic encoder is compromised, covering nearly all aspects of steganalysis.

E. Chosen Plaintext Attack

Consider a public-key cryptography system $CS = (G, E, D)$ and a chosen plaintext attacker $\mathcal{A}$. $\mathcal{A}$ is allowed to play a game described as follows:

- **Key generation stage.** $(PK, SK) \leftarrow G(1^k)$.
- **Learning stage.** $\mathcal{A}$ sends plaintext $m_{\mathcal{A}}$ to the oracle and returns $E(PK, m_{\mathcal{A}})$. $\mathcal{A}$ can perform this stage multiple times.
- **Challenge stage.** $\mathcal{A}$ sends hiddentext $m \in \mathcal{A} \setminus \{m_{\mathcal{A}}\}$ to the oracle, which will flip a coin $b \in \{0, 1\}$. If $b = 0$, $\mathcal{A}$ obtains $c = E(PK, m)$; If $b = 1$, $\mathcal{A}$ obtains $u \leftarrow U_{|E(PK,\cdot)|}$.
- **Guess stage.** $\mathcal{A}$ output a bit b' as a "guess" about whether it obtains a plaintext or a random string.

Define the Chosen Plaintext Attack (CPA) advantage of $\mathcal{A}$ against S by:

$$\mathbf{Adv}^{\text{cpa}}_{CS}(\mathcal{A}, k) = \left| \begin{array}{l} \Pr\limits_{PK}[\mathcal{A}(PK, c) = 1] \\ -\Pr\limits_{PK}[\mathcal{A}(PK, u) = 1] \end{array} \right|. \tag{7}$$

Define the *insecurity* of CS over by

$$\mathbf{InSec}^{cpa}_{CS}(t, l, k) = \max_{\mathcal{A} \in \mathcal{A}_{(t,l)}} \{\mathbf{Adv}^{cpa}_{CS}(\mathcal{A}, k)\}, \tag{8}$$

$\mathcal{A}_{(t,l)}$ denotes the set of all adversaries that send at most $l(k)$ bits and run in time $t(k)$. $l(k)$ and $t(k)$ are polynomials of k.

A public-key encryption system that is *indistinguishable from uniformly random bits under chosen plaintext attack (IND$-CPA)* if $\mathbf{InSec}^{cpa}_{CS}(t, l, k)$ is negligible in k, i.e., no PPT adversary can distinguish encryption results and uniform random strings with nonnegligible probability.

F. Pseudorandom Elliptic Curve Point Encoding

Typical elliptic curve public-key encryption schemes are not IND$-CPA secure because plaintext points possess unique algebraic properties that make them identifiable.

For instance, given suspected points $(x, y) \in F_p$ on the curve $E_{A,B} : By^2 = x^3 + Ax^2 + x \ (mod\ p)$, a detector can ascertain if they lie on the curve by verifying the equation. Similarly, if a detector has suspected coordinate $x \in F_p$, they can compute $(x^3 + Ax + B)$ and examine whether the result is a quadratic residue of the field F_p in polynomial time. Since the quadratic residue of field F_p contains $\frac{p-1}{2}$ elements, transmitting untreated elliptic curve points directly during covert communication is perilous.

To address this issue, many efforts have been made including Elligator2 [37]. This method comprises a pair of bijection functions (ϕ ψ) used to encode elliptic curve points into a uniform random string and to decode a string into elliptic curve points. All calculations below are performed in the F_p field, which we will not emphasize further.

Let χ define the Legendre sign of field F_p, i.e.,

$$\chi : F_p \to F_p, \quad \chi(a) = a^{\frac{p-1}{2}}$$

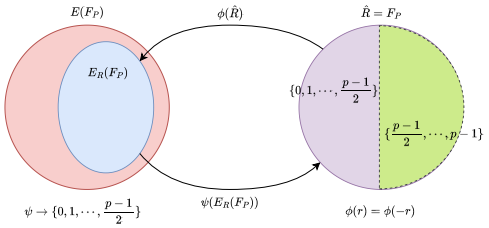


Fig. 3. The relation of encode function and decode function.

Let the square root for the set of quadratic residue F_p^2 be a single-valued function, i.e.,

$$\sqrt{x} : F_p^2 \to \{0, 1, \ldots, \frac{p-1}{2}\}$$

Suppose that the elliptic curve over F_p satisfies the following forms:

$$E_{A,B} : y^2 = x^3 + Ax^2 + Bx,$$
$$p \equiv 1(mod\ 4),\ \ \chi(A^2 - 4B) = -1,$$
$$AB(A^2 - 4B) \neq 0.$$

Arbitrarily select $u \in F_p$ satisfying $\chi(u) = -1$.
Define *Decode Function Feasible Region* $R \subseteq F_p$:

$$R = \{r \in F_p : r \neq 0,\ 1 + ur^2 \neq 0,\ A^2ur^2 \neq B(1 + ur^2)^2\}.$$

Define *Decode Function* $\phi : R \to E(F_p)$:

$$v = \frac{-A}{1 + ur^2}, \tag{9}$$
$$\epsilon = \chi(v^3 + Av^2 + Bv), \tag{10}$$
$$x = \epsilon v - \frac{(1 - \epsilon)A}{2}, \tag{11}$$
$$y = -\epsilon\sqrt{x^3 + Ax^2 + Bx}. \tag{12}$$

Define *Encode Function Feasible Region* $E_R(F_p) \subseteq E(F_p)$:

$$E_R(F_p) = \{(x, y) | (x, y) \in E(F_p)$$
$$x \neq -A;\ \ y = 0 \Rightarrow x = 0;\ \ \chi(-ux(x + A)) = 1\}.$$

Define *Encode Function* $\psi : E_R(F_p) \to F_p$:

$$\psi((x, y)) = \bar{r} = \begin{cases} \sqrt{\dfrac{-x}{(x + A)u}}, & if\ y \in \sqrt{F_p^2} \\ \sqrt{\dfrac{-(x + A)}{ux}}, & if\ y \notin \sqrt{F_p^2}. \end{cases} \tag{13}$$

As shown in Fig.3 and Theorem 6 (see Appendix), the functions (ϕ ψ) create a bijective function relationship between approximately half of the elliptic curve points $E_R(F_p)$ and uniform random strings $\{0, 1, \cdots, \frac{p-1}{2}\}$.

Choosing $p = 2^k - e$, the likelihood of distinguishing ψ encoding output from a truly random $(k-1)$-bit string is $1 - \frac{(p+1)/2}{2^{k-1}} \leq \frac{e}{2^k}$, which is negligible in parameter k. Therefore, the output of the encoding function is indistinguishable from a uniformly random string.

In this paper, it is crucial to emphasize that Elligator2 serves solely as a point obfuscation technique, with the capability of covering curve points of arbitrary order. As a result, relying on Elligator2 alone does not ensure that encryption

and decryption operations are performed within the large prime order subgroup, which consequently cannot guarantee the validity of the Decisional Diffie-Hellman assumption in the elliptic curve group (ECDDH). Therefore, to construct an IND$-CPA secure public-key encryption system, we develop a public-key cryptography scheme that combines the Elligator2 method with additional measures to ensure the satisfaction of the Elliptic Curve Decisional Diffie-Hellman (ECDDH) problem criteria.

IV. Our Proposed Method

The architecture of our proposed public-key steganography system framework is illustrated in Fig.4. We will outline the thought process behind constructing this system as follows.

According to Hopper's theory [34], to create a public-key steganography system that can withstand chosen hiddentext attacks, it is necessary to build a public-key encryption system that is indistinguishable from uniformly random bits under chosen plaintext attack (IND$-CPA). To achieve this, we propose a public-key encryption system based on the Elligator2 technique that meets the requirements of the Elliptic Curve Decisional Diffie-Hellman (ECDDH) problem.

Using the above public-key encryption system, we build a public-key steganography system based on reversible mapping and sampling. For the public-key steganography construction designed in our paper under the chosen hiddentext attack, we provide comprehensive proof of its security.

In addition, we construct an instance of a public-key steganography system based on a text-to-speech model, WaveGlow. By introducing reversible mapping on zero-mean spherical Gaussian sampling, we are able to generate highly naturalistic audio that can hide a message and thus put our construction into practice.

In the implementation, we deploy the entire system on Curve25519, which satisfies all the necessary parameter constraints described in Sec III-F for using the Elligator2 technique.

A. Pseudorandom Public-Key Encryption Construction Using Elligator2

We propose a public-key encryption construction that is IND$-CPA secure. The construction consists of two hybrid encryption constructions. First, suppose that the sender and the receiver share a base point B_0 of elliptic curve order. We obtain the first ciphertext by uniformly sampling points on the entire curve using B_0 and its scalar multiplication and apply the Elligator2 [37] encoding function, yielding a binary string that is indistinguishable from a uniformly random string. Next, to satisfy the elliptic curve Decisional Diffie-Hellman assumption, the chosen point will be multiplied by a fixed constant factor to map it to a curve point of order a large prime Q, following which the resulting point is hashed to generate the key for the second ciphertext. Furthermore, the message is encrypted with an IND$-CPA secure symmetric encryption construction to obtain the second ciphertext. By combining these two encryption constructions, we achieve IND$-CPA security for the entire encryption construction.

Here, we introduce the details of our proposed public-key encryption construction using Elligator2. Let $E_{A,B}(x, y)$ be an elliptic curve defined over F_p that satisfies the following conditions: $q \equiv 1 \pmod 4$, $\chi(A^2 - 4B) = -1$, and $AB(A^2 - 4B) \neq 0$. Suppose $H : \{0,1\}* \to \{0,1\}^\kappa$ denotes the hash function, $\kappa \leq k = |p|$. $E_{(\cdot)}(\cdot)$, $D_{(\cdot)}(\cdot)$ denote the encryption and decryption functions of a symmetric encryption construction satisfying IND$-CPA, keyed by κ-bit keys. Easy to find such symmetric encryption construction (such as AES).

Assume that the order of $E_{A,B}(x, y)$, $|E_{A,B}(F_p)|$, satisfies $|E_{A,B}(F_p)| = rQ$, where Q is a sufficiently large prime and $(r, Q) = 1$. Let $B_0 \in E_{A,B}(F_p)$, where $\text{Order}(B_0) = rQ$. Then, we compute $B' = r \cdot B_0$, hence $\text{Order}(B') = Q$. B' generates the subgroup of $|E_{A,B}(F_p)|$ whose order is Q, thus the ECDDH assumption holds.

Assuming that $(PK, SK) = (x \cdot B', x)$, we present our pseudorandom public-key encryption algorithm E and decryption algorithm D through Algorithm 1 and Algorithm 2, respectively.

Algorithm 1 Elliptic Curve Pseudorandom Public-Key Encryption (E)

INPUT: $m \in \{0,1\}^*, B_0, B', r, Q, PK = x \cdot B'$
OUTPUT: c

repeat
  $a \leftarrow U(0, rQ)$
  $V = a \cdot B_0$
until $V \in E_R(F_p)$
$K = H(a \cdot PK) = H(a \cdot x \cdot B')$
$c_1 = \psi(V), c_2 = E_K(m), c = c_1 || c_2$
return c

Algorithm 2 Elliptic Curve Pseudorandom Public-Key Decryption (D)

INPUT: $c, B_0, B', r, Q, SK = x$
OUTPUT: m

separate c into c_1, c_2
$V = \phi(c_1)$
$K = H(a \cdot x \cdot B') = H(a \cdot x \cdot r \cdot B_0) = H(x \cdot r \cdot V)$
$m = D_K(c_2)$
return m

The output of the encryption algorithm consists of 2 parts, c_1 and c_2. c_1 is a product of Elligator2's encode function ψ such that the ECDDH assumption holds. Therefore, it is IND$-CPA secure. c_2 is a result of a reliable symmetric encryption construction and is assumed to be secure against CPA. As a result, the overall construction is IND$-CPA secure, with $\mathbf{InSec}_{CS}^{cpa}(t, l, k)$ negligible in k.

B. Public-Key Steganography System Based on a Generative Model

For generative models with accessible probability distributions, it is essential to emphasize that our proposed steganographic method, based on reversible mapping, is specifically

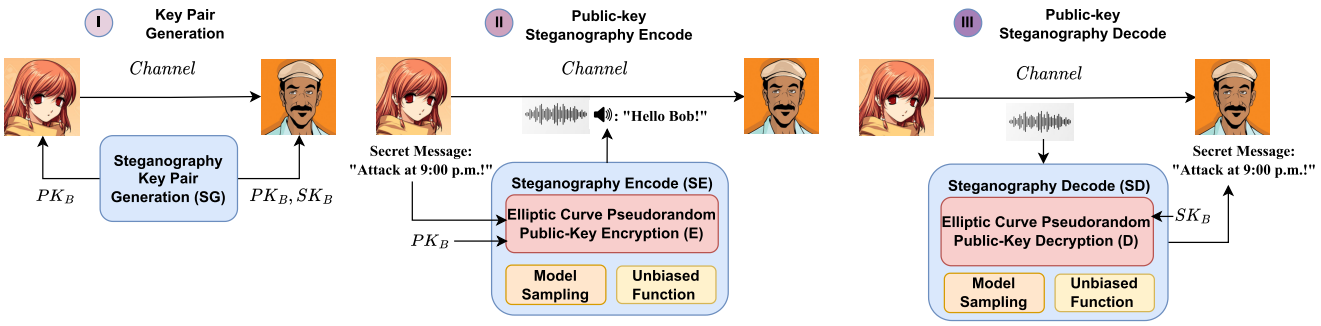


Fig. 4. The framework of our public-key steganography system.

designed for such models. These generative models, exemplified by autoencoders [51], generative adversarial networks (GANs) [15], invertible neural networks (INNs) [58], and others, provide users with access to either the input data or the generated data's probability distribution. Our method is tailored to leverage this accessibility, ensuring both high efficiency and provable security. These models can fit natural data distributions into one or multiple latent variable distributions during training and allow for sampling from these distributions in the generation process.

Let C_h be the explicit data distribution of the generative model when given history h. Let $f_h : m \in \{0,1\}^q \to C_h^m \subseteq C_h$ be a reversible function from an q-bit binary string to a certain subdistribution, where $\bigcup_{m\in\{0,1\}^q} C_h^m = C_h$ and $\forall\, m_1 \neq m_2, C_h^{m_1} \cap C_h^{m_2} = \emptyset$. Reversibility means we can calculate the binary string by $f_h^{-1} : C_h^m \to \{0,1\}^q$.

We say f_h is perfectly unbiased if $\forall\ m \in \{0,1\}^q$ and h, $\Pr(C_h^m) = \frac{1}{2^q}$, which means f_h reversibly maps the q-bit binary string to 2^q mutually exclusive and measure-equivalent subdistributions.

We say f_h is ϵ-biased if $\forall\, m \in \{0,1\}^q$ and h, $|\Pr(C_h^m) - \frac{1}{2^q}| = \epsilon(k)$, where ϵ is a function of parameter k.

Combining reversible mapping and our pseudorandom public-key encryption construction, we propose our construction of public-key steganography system $SS = (SG, SE, SD)$. $SG(1^k)$ takes a random bitstring as input and generates a key pair (PK, SK). $SE(PK, m, h)$ takes the public key, hidden text m and distribution of the history-based channel C_h, and outputs a stegotext s sampled from C_h. $SD(SK, s, h)$ takes the private key, stegotext and history, and outputs the hidden text m. Here, we introduce the details of our proposed public-key steganography construction. Algorithms 3, 4, and 5 presented below illustrate the specific procedures for SG, SE, and SD, respectively.

Suppose we have selected curve $E_{A,B}(x, y)$ over F_P, $|P| < k$ satisfying $E_{A,B} : y^2 = x^3 + Ax^2 + Bx$, $p \equiv 1(mod\ 4)$, $\chi(A^2 - 4B) = -1$, $AB(A^2 - 4B) \neq 0$, whose order is N, and N has a sufficiently large prime factor Q, $N = rQ, (Q, r) = 1$. We can easily find $u \in F_p$ satisfying $\chi(u) = -1$. Assume that the reversible function f_h is ϵ-biased when ϵ is negligible in k.

In SG, as described in Algorithm 3, the secret key x is repeatedly selected, and the corresponding public key is calculated on a prime-order group generated by the base point B'. This process continues until the public key can be encoded by using the Elligator2 technique.

Algorithm 3 Steganography Key Pair Generation (SG)

INPUT: $1^k \in U(|k|)$
OUTPUT: PK, SK

Given $E_{A,B}(x, y)$,find base point B_0 : Order$(B_0) = N$.
Computing $B' = r \cdot B_0$.
repeat
  $x \leftarrow U(0, Q)$, $V = x \cdot B'$
until $V \in E_R(F_P)$
$PK = V, SK = x$

Algorithm 4 Steganography Encode (SE)

INPUT: $m \in \{0,1\}^*, B_0, B', r, Q, PK = x \cdot B', h, f, G$
OUTPUT: s

repeat
  $a \leftarrow U(0, rQ)$, $V = a \cdot B_0$
until $V \in E_R(F_p)$
$K = H(a \cdot PK) = H(a \cdot x \cdot B')$
$c_1 = \psi(V), c_2 = E_K(m), c = c_1||c_2, s_0 = \{\}$
$n = length(c), i = 0$
while $i < n$ **do**
  $x = c[i : i + q]$
  $C_h \leftarrow G(h)$, $C_h^x = f_h(x, C_h)$
  $s \leftarrow C_h^x \subset C_h$
  append s to s_0, append s to h, $i = i + q$
end while
return s_0

In SE, as described in Algorithm 4, we encrypt the concealed text by using the pseudorandom public-key encryption construction that we have proposed. Subsequently, we group the data into segments of q bits each and apply reversible mapping to map them onto a unique subdistribution of data. From this unique subdistribution, we proceed to randomly sample the stego.

In SD, as described in Algorithm 5, we take advantage of the reversibility of the mapping to extract the ciphertext from the subdistribution where the stego is located. Subsequently, we decrypt the ciphertext by using the pseudorandom public-key encryption construction.

Algorithm 5 Steganography Decode (SD)

INPUT: $s, B_0, B', r, Q, SK = x, C_h, f$
OUTPUT: m
$c = \{\}$
for each $x \in s_0$ **do**
  $C_h \leftarrow G(h)$
  $c = c \mid\mid f_h^{-1}(x, C_h)$
end for
separate c into c_1, c_2
$V = \phi(c_1)$
$K = H(a \cdot x \cdot B') = H(a \cdot x \cdot r \cdot B_0) = H(x \cdot r \cdot V)$
$m = D_K(c_2)$
return m

In essence, our approach begins by designing a public-key algorithm in SE and SD that achieves IND$-CPA. In practical scenarios involving Artificial Intelligence-Generated Content (AIGC), where the channel is inundated with content from various generative models, we utilize a pre-trained white-box generative model. This model provides us with a probability distribution for either the input or the generated data, enabling the construction of an unbiased function. For creating the cover, we employ random sampling with the white-box generative model, driven by random bit strings. When generating stego content, we ensure that the sampling process is guided by the public key-encrypted ciphertext and the unbiased function. This way, we reduce the challenge of distinguishing stego from cover in a Channel to distinguishing public key-encrypted ciphertext from random bit strings, a task already guaranteed by the security properties of the public key algorithm IND$-CPA.

Here comes a rigorous proof. Given the steganography system $SS = (SG, SE, SD)$, and the algorithm definition is shown above, we sketch the proof of security of SS against CHA.

Theorem 1: $\mathbf{InSec}^{cha}_{SS,C}(t, l, k)$ *is negligible in* k.

Proof: Assume that there exists a PPTM $\mathcal{A}$ that achieves a nonnegligible advantage $\mathbf{Adv}^{cha}_{SS,C}(\mathcal{A}, k) = |\Pr[\mathcal{A}(PK, s) = 1] - \Pr[\mathcal{A}(PK, c) = 1]|$ in distinguishing between the output of algorithm SE and the generated data sampled from $C_h^m \subseteq C_h$. Furthermore, let us assume that $\mathcal{A}$ can question an oracle, which means that $\mathcal{A}$ is allowed to access the same generated model. We will construct a program $\mathcal{A}'$ that plays the chosen-plaintext attack against the public-key encryption system $CS = (G, E, D)$,where we define E and D as before and use SG as G. The goal of $\mathcal{A}'$ is to distinguish plaintext $E_{PK}(m_{\mathcal{A}})$ from uniform random string $U_{|E(PK,\cdot)|}$.

$\mathcal{A}'$ first chooses history $h_{\mathcal{A}}$ and a message $m_{\mathcal{A}}$ and then runs $\mathcal{A}$ to go through the key generation stage and learning stage to obtain a well-trained PPTM $\mathcal{A}$ that achieves $\mathbf{Adv}^{cha}_{SS,C}(\mathcal{A}, k) = |\Pr[\mathcal{A}(PK, s) = 1] - \Pr[\mathcal{A}(PK, c) = 1]|$ is nonnegligible in k. During the Challenge stage, $\mathcal{A}'$ picks plaintext $m \in \mathcal{A} \setminus \{m_{\mathcal{A}}\}$ and sends it to the oracle. The oracle will flip a coin b, where for $b = 0$, $\mathcal{A}'$ obtains $c = E_{PK}(m)$, and for $b = 1$, $\mathcal{A}'$ obtains $u \leftarrow U_{E(PK,\cdot)}$. After receiving the oracle's return, $\mathcal{A}'$ encodes it into multimedia data using the generated model and sends it to W to let it guess. $\mathcal{A}$ outputs a bit b' as its answer, which is also $\mathcal{A}'$'s answer. The total time of the whole process is $t + \mathcal{O}(lk)$.

Since the reversible mapping f is ϵ-biased, we have $|\Pr[\mathcal{A}'(PK, c) = 1] - \Pr[\mathcal{A}(PK, SE(PK, m, h_{\mathcal{A}})) = 1]| \leq \lceil \frac{|m|}{q} \rceil \cdot \epsilon$ and $|\Pr[\mathcal{A}'(PK, u) = 1] - \Pr[\mathcal{A}(PK, c \leftarrow C_h) = 1]| \leq \lceil \frac{|u|}{q} \rceil \cdot \epsilon$. Since $\mathbf{Adv}^{cpa}_{CS}(\mathcal{A}', k) = |\Pr[\mathcal{A}'(PK, c) = 1] - \Pr[\mathcal{A}'(PK, u)] = 1|$, we have $\mathbf{Insec}^{cpa}_{CS}(t + \mathcal{O}(lk), l, k) = \max_{\mathcal{A} \in \mathcal{A}_{(t,l)}}\{\mathbf{Adv}^{cpa}_{CS}(\mathcal{A}, k)\} \geq \mathbf{Adv}^{cha}_{SS,C}(\mathcal{A}, k) - \max\{\lceil \frac{|m|}{q} \rceil, \lceil \frac{|u|}{q} \rceil\} \cdot \epsilon$, which is nonnegligible in k. This contradicts the proposition $\mathbf{Insec}^{cpa}_{CS}(t, l, k)$ is negligible in k.

Therefore, $\mathbf{Insec}^{cha}_{SS,C}(t, l, k)$ is negligible in k. □

TABLE I
THE EXPECTED RUNNING TIME OF PSEUDORANDOM PUBLIC-KEY SCHEME

	Encrypt (E)	Decrypt (D)
RSA-based method [23]	$\mathcal{O}(\|m\| \cdot log(N))$	$\mathcal{O}(\|m\| \cdot log(N))$
Elgamal-based method [23]	$\mathcal{O}(\|P_{EG}\| + \|m\|)$	$\mathcal{O}(\|P_{EG}\| + \|m\|)$
Our ECC-base method	$\mathcal{O}(\|P_{ECC}\| + \|m\|)$	$\mathcal{O}(\|P_{ECC}\| + \|m\|)$

C. Computational Complexity and Embedding Efficiency

Assuming $|m|$ is the length of the message, N is the large integer chosen for RSA, P_{EG} is the modulus for ElGamal, and P_{ECC} is the modulus for ECC. The complexity of these algorithms is presented in Table I:

The encryption result for these three methods not only includes the ciphertext bits but also an additional term. The length of this additional term is influenced by the security parameter settings of the encryption algorithms, such as the bit length of the large integer N in RSA, the bit length of the selected prime P_{EG} in Elgamal, and the bit length of the selected prime P_{ECC} in ECC. This becomes a bottleneck, impacting both communication and computational efficiency. When these three algorithms operate at the same security level (e.g., 128 security bits), ECC requires a significantly smaller parameter length (256) compared to RSA (3,072) and Elgamal (3,072). In summary, due to its smaller parameter length requirement at an equivalent security level, ECC outperforms other algorithms in the same class in terms of both computational complexity and embedding efficiency.

D. An Instance of Public-Key Steganography Based on WaveGlow

WaveGlow is a reversible generative model constructed using affine coupling layers. It generates samples from a zero-mean spherical Gaussian, which have the same number of dimensions as the intended output. During inference, given a Gaussian latent variable $z \sim \mathcal{N}(z; 0, I)$ and the mel-spectrogram of a paragraph of certain text, WaveGlow computes the distribution of audio samples conditioned on the mel-spectrogram:

$$x = f_0 \circ f_1 \circ \ldots \circ f_k(z, F_{mel}),$$

where f_i denotes the i-th invertible affine coupling layer.

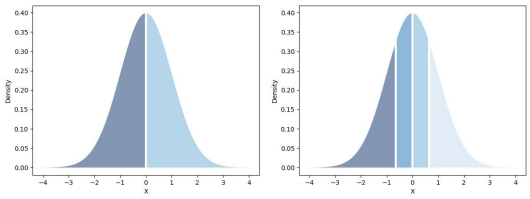


Fig. 5. Examples of mapping ($q = 1, 2$ bits).

Assuming that two communicating parties share the same spectrogram generation model (SPN) and a pretrained WaveGlow model, we design a reversible mapping from a uniformly random q-bit binary message m to a Gaussian latent variable sample unbiasedly:

$$f : m \rightarrow (\alpha_-, \alpha_+) \subset \mathcal{N}(z; 0, I),$$
$$z \leftarrow (\alpha_-, \alpha_+),$$

where $\alpha_- = F^{-1}(\frac{m}{2^q})$, $\alpha_+ = F^{-1}(\frac{m+1}{2^q})$ and F^{-1} denote the inverse of cumulative distribution function (CDF) of the Gaussian distribution.

Consider Fig.5 as an example. We partition the Gaussian distribution into 2^q segments, each with an equal probability. This guarantees that when m is selected uniformly, it falls within every interval with identical probability.

Building upon this mapping, we present an instance of public-key steganography based on WaveGlow, wherein the Gaussian distribution is transformed into the desired distribution through a series of invertible layers using the mel-spectrogram and the generated samples. First, given a text, the SPN transforms it into a mel-spectrogram F_{mel}. Afterward, we map the binary message to a series of Gaussian latent variables. Finally, the mel-spectrogram F_{mel} and a series of Gaussian latent variables z are input into WaveGlow to generate stego audio x. The specific procedures for the SE and SD processes are illustrated in Algorithm 6 and Algorithm 7, respectively, as presented below.

For each bit fragment, SE maps it to a specific interval of the Gaussian distribution. The stego is sampled from this subdistribution. Since the mapping is unbiased and the bit string is assumed to be pseudorandom, the distribution of the stego is indistinguishable from the Gaussian distribution.

The decoding process SD is simple. After the receiver obtains the text from the stego audio, the same mel spectrogram F_{mel} is obtained by using the SPN. By using the reversibility of WaveGlow, the latent variable z is obtained. Then the receiver checks the interval where z falls and obtains m. This WaveGlow-based public-key steganography instance can be efficiently deployed in environments such as Discord, an online gaming community platform that incorporates a variety of AI-generated services for text, image, audio, and video where the use of generated audio seamlessly integrates into the user experience, remaining both permissible and unobtrusive.

Given the steganography system $SS = (SG, SE, SD)$ based on WaveGlow, and the algorithm definition shown above, we sketch the proof of security of SS against the Chosen-Hiddentext attack.

Algorithm 6 Public-Key Steganography Encode (SE) Based on WaveGlow

INPUT: $m \in \{0, 1\}^*$, B_0, B', r, Q, $PK = x \cdot B'$, text
OUTPUT: s

repeat
  $a \leftarrow U(0, rQ)$, $V = a \cdot B_0$
until $V \in E_R(F_p)$
$K = H(a \cdot PK) = H(a \cdot x \cdot B')$
$c_1 = \psi(V), c_2 = E_K(m), c = c_1||c_2$
$z_0 = \{\}$, $n = length(c), i = 0$
while $i < n$ **do**
  For bit fragment $c[i : i + q]$:
    $m_c = \Sigma_{j=0}^{j=q-1} c[i + j] * 2^j$
    $\alpha_- = F^{-1}(\frac{m_c}{2^q}), \alpha_+ = F^{-1}(\frac{m_c+1}{2^q})$
    sample z from $(\alpha_-, \alpha_+) \subset \mathcal{N}(z; 0, I)$
    append z to z_0, $i = i + q$
end while
$F_{mel} =$ SPN(text)
$audio =$ WaveGlow(F_{mel}, z_0)
return $audio$

Algorithm 7 Public-Key Steganography Decode (SD) Based on WaveGlow

INPUT: $audio$, B_0, B', r, Q, $SK = x$, text
OUTPUT: m

$F_{mel} =$ SPN(text)
$z_0 =$ WaveGlow$^{-1}(F_{mel}, audio)$, $c = \{\}$
for each $z \in z_0$ **do**
  $c = c \;||\; \lfloor F(z) * 2^q \rfloor$
end for
separate c into c_1, c_2
$V = \phi(c_1)$
$K = H(a \cdot x \cdot B') = H(a \cdot x \cdot r \cdot B_0) = H(x \cdot r \cdot V)$
$m = D_K(c_2)$
return m

Theorem 2: $\mathbf{InSec}^{cha}_{SS,C}(t, l, k)$ *is negligible in* k.

Proof: Assume that there exists a PPTM $\mathcal{A}$ with a nonnegligible advantage $\mathbf{Adv}^{cha}_{SS,C}(\mathcal{A}, k) = |\Pr[\mathcal{A}(PK, s) = 1] - \Pr[\mathcal{A}(PK, c) = 1]|$ in distinguishing between the output of algorithm SE and randomly generated audio whose latent variable is sampled from a standard Gaussian distribution. Additionally, suppose $\mathcal{A}$ can question an oracle, which means that $\mathcal{A}$ is allowed to access the same WaveGlow and SPN model. Construct a program $\mathcal{A}'$ that plays the chosen-plaintext attack against the public-key encryption system $CS = (G, E, D)$.

During the challenge stage, $\mathcal{A}'$ selects a plaintext $m \in \mathcal{A} \setminus \{m_{\mathcal{A}}\}$ and sends it to the oracle. The oracle will flip a coin b. For $b = 0$, $\mathcal{A}'$ obtains $c = E_{PK}(m)$. For $b = 1$, $\mathcal{A}'$ obtains $u \leftarrow U_{E(PK,\cdot)}$.

After receiving the oracle's return, $\mathcal{A}'$ encodes it into audio using WaveGlow and sends it to $\mathcal{A}$ to let it guess. $\mathcal{A}$ outputs a bit b' as its answer, which is also $\mathcal{A}'$'s answer. The total time of the whole process is $t + \mathcal{O}(lk)$.

Since WaveGlow uses unbiased mapping and the encoding procedure is reversible, we have $\Pr[\mathcal{A}'(PK, c) = 1] = \Pr[\mathcal{A}(PK, SE(PK, m, h_{\mathcal{A}})) = 1]$ and $\Pr[\mathcal{A}'(PK, u) = 1] = \Pr[\mathcal{A}(PK, c \leftarrow C_h) = 1]$. Thus, $\mathbf{Adv}^{cpa}_{CS}(\mathcal{A}', k) = |\Pr[\mathcal{A}'(PK, c) = 1] - \Pr[\mathcal{A}'(PK, u)] = 1| = \mathbf{Adv}^{cha}_{SS}(\mathcal{A}, k)$. $\mathbf{Insec}^{cpa}_{CS}(t + \mathcal{O}(lk), l, k) = \max_{\mathcal{A}\in\mathcal{A}_{(t,l)}}\{\mathbf{Adv}^{cpa}_{CS}(\mathcal{A}, k)\} \geq \mathbf{Adv}^{cha}_{SS,C}(\mathcal{A}, k)$ is nonnegligible in k. This contradicts the proposition that $\mathbf{Insec}^{cpa}_{CS}(t, l, k)$ is negligible in k.

Therefore, $\mathbf{Insec}^{cha}_{SS,C}(t, l, k)$ is negligible in k. □

E. Implement Public-Key Steganography on Curve25519

We use Curve25519 for the entire system, which is an elliptic curve used in elliptic-curve cryptography; it offers 128 bits of security (256-bit key size). Additionally, it is one of the fastest curves in ECC. The parameter specifications are shown below.

$$\text{Curve25519}\begin{cases} p = 2^{255} - 19, \\ y^2 = x^3 + 486662x^2 + x, \\ A = 486662,\ \ B = 1, \\ \|E(F_p)\| = 2^{255} + e, e < 2^{128}, \\ \|E(F_p)\| = 8P,\ P = 2^{252} + \dfrac{e}{8} \text{ large prime}, \\ \text{Base point } B' : x = 9. \end{cases}$$

Given that the parameters satisfy $p \equiv 1(mod\ 4)$, $\chi(A^2 - 4B) = -1$, $A \neq 0$, $B \neq 0$, $A^2 - 4B \neq 0$. We select $u = 2$ such that $\chi(u) = -1$. Since $\frac{\|E_R(F_q)\|}{\|E(F_q)\|} \approx \frac{1}{2}$, approximately half the points on the curve can be encoded into random strings.

V. Steganographic Key Exchange

Public-key steganography can be effectively utilized for key transmission. Once a connection is established, more efficient symmetric steganography can be employed for communication. Moreover, for communication entities that lack public keys, a steganographic key exchange approach can be adopted.

The steganographic key exchange involves transmitting a sequence of messages that simulate typical communication traffic between Alice and Bob, ultimately using these messages to establish a shared key. The success of this method depends on whether the shared key is indistinguishable from a random key by the warden. Alice and Bob can then use their shared key with confidence in a secret-key system.

Our proposal is a single-round asynchronous algorithm based on the Elliptic Curve Diffie-Hellman (ECDH) for covert key exchange, as shown in the following algorithm (taking Alice's node as an example).

To negotiate the session key K, both participants in communication must perform one round of encoder operations, transmit the stego, and then perform independent decoder operations. The session key is derived from the content being communicated.

The security of this method is straightforward since the public key pk generates a point on a large prime order subgroup and is mapped to a uniform random string. Thus, it is difficult to distinguish between the stego generated and to compute the private key sk.

Algorithm 8 Steganographic Key Exchange Encoder

INPUT: $B_0, B', r, Q, text$
OUTPUT: s
1: **repeat**
2: $\quad sk_A \leftarrow U(0, rQ)$
3: $\quad pk_A = sk_A \cdot B_0$
4: **until** $pk_A \in E_R(F_p)$
5: $c = \psi(pk_A)$
6: $s =$ WaveGlow_Encoder$(c, text)$
7: **return** s

Algorithm 9 Steganographic Key Exchange Decoder

INPUT: $s, B_0, B', r, Q, pk_B, x_A, text$
OUTPUT: K
$c =$ WaveGlow_Decoder$(s, text)$
2: $pk_B = \phi(c)$
$K = x_A \cdot r \cdot pk_B$
4: **return** K

TABLE II
Statistical Test for Pseudorandomness of Encryption Algorithm E

NIST SP 800-22	$n = 100$	$\alpha = 0.01$	
Statistical Test	P-VALUE	PROPORTION	Result
Frequency	0.319084	98/100	PASS
BlockFrequency	0.534146	99/100	PASS
CumulativeSums	0.275709	96/100	PASS
Runs	0.062821	100/100	PASS
LongestRun	0.026948	100/100	PASS
Rank	0.040108	100/100	PASS
FFT	0.779188	100/100	PASS
NonOverlappingTemplate	0.637119	99/100	PASS
OverlappingTemplate	0.350485	100/100	PASS
Universal	0.474986	99/100	PASS
ApproximateEntropy	0.719747	100/100	PASS
RandomExcursions	0.134686	62/62	PASS
Serial	0.096578	98/100	PASS
LinearComplexity	0.437274	98/100	PASS

VI. Experiments

In this section, we evaluate the pseudorandomness of the proposed elliptic curve pseudorandom public-key encryption algorithm through statistical tests. Additionally, we validate the security of our proposed public-key steganography instance based on WaveGlow through steganalysis experiments.

A. Statistical Test for Pseudorandomness

To test the pseudorandomness of our elliptic curve pseudorandom public-key encryption algorithm, we employed the NIST SP 800-22 test suite.

First, we generated a key pair and randomly selected plaintext m to obtain the encryption result $c = E_{PK}(m)$. We repeated this process and accumulated the results in binary string form until we obtained more than 10^8 bits. The string was then divided into 100 streams of equal length to conduct 15 statistical tests. We repeated the process with different key pairs multiple times and obtained similar outcomes. Finally, we select one trial and present the results below:

To test the randomness of the encrypted data, we computed the proportion of sequences that passed a particular statistical test. With a significance level of $\alpha = 0.01$ and $n = 100$ sequences, we calculated the acceptable range of proportions by using the confidence interval formula $\hat{p} \pm 3\sqrt{\frac{\hat{p}(1-\hat{p})}{n}}$, where $\hat{p} = 1-\alpha$. If the proportion falls outside of this interval, then there is evidence that the data are nonrandom.

For $n = 100$ and $\alpha = 0.01$, the calculated confidence interval is $0.99 \pm 3\sqrt{\frac{\hat{p}(1-\hat{p})}{n}} = 0.99 \pm 0.0298$ (i.e., the proportion should be greater than 0.9602).

Based on Table II, our designed public-key encryption algorithm passed all 15 types of tests in the NIST SP 800-22 suite, ensuring that its encrypted output demonstrates pseudorandomness as demonstrated by the experimental results.

B. Steganalysis Experiments

Steganalysis is a technology for distinguishing stegotext from covertext, the essence of which is a binary classifier.

$$\mathbf{F}(X) = \begin{cases} 0, & \text{if } \Phi(X) < 0.5, \\ 1, & \text{if } \Phi(X) \geq 0.5, \end{cases} \tag{14}$$

where $\Phi(X) \in [0, 1]$ is the probability that the input X is a cover ($\mathbf{F} = 0$) or a stego ($\mathbf{F} = 1$). The false alarm occurs when X is a cover audio while $\mathbf{F} = 1$, and missed detection occurs when X is a stego audio while $\mathbf{F} = 0$. False alarm and miss detection are, respectively defined as

$$P_{\text{FA}} = \Pr\{\mathbf{F}(X) = 1 \mid X \in \mathcal{C}\}, \tag{15}$$

$$P_{\text{MD}} = \Pr\{\mathbf{F}(X) = 0 \mid X \in \mathcal{S}\}. \tag{16}$$

where $\mathcal{C}$ and $\mathcal{S}$ are the cover set and the stego set respectively. Then the total performance is the probability of detection error computed from P_{FA} and P_{MD} as follows.

$$P_{\text{E}} = \frac{P_{\text{FA}} + P_{\text{MD}}}{2}. \tag{17}$$

We randomly selected various short sentences and transformed them into mel-spectrograms using the SPN networks. By using the generated key pair via SG and the algorithm SE, we encoded a specific message into audio, creating the stegotext set. In addition, we generated normal audio from identical short text sentences to establish the covertext set.

We employ three advanced audio steganography techniques (two based on machine learning and one based on deep learning), namely Liu et al.'s [43], Luo et al.'s [41] and Lin et al.'s [59]. Liu et al.'s method relies on the audio second derivative of the mel-frequency cepstral coefficients (MFCC) feature detection method. Luo et al.'s method uses a detection method based on the frequency-domain MFCC feature of the second-order difference residual signal and the time-domain Markov transition probability feature of the original audio signal. Lin et al.'s approach is based on CNN's audio steganography method, with a carefully designed convolutional layer. The steganalysis experiments were conducted on a dataset consisting of 10000 samples of stegotext and covertext, which were divided into a training set, validation set, and test set in different ratios.

TABLE III
STEGANALYSIS EVALUATION OF LIU ET AL.'S AND LUO ET AL.'S METHODS FOR STEGANOGRAPHIC ALGORITHMS (SE)

	Ratio (TRN:TST)	P_{E}	P_{FA}	P_{MD}
Liu et al.'s [42]	8:2	0.4971	0.4944	0.4999
	7:3	0.4973	0.4960	0.4986
	6:4	0.4959	0.4976	0.4941
	5:5	0.4977	0.4990	0.4963
Luo et al.'s [41]	8:2	0.5010	0.4900	0.5120
	7:3	0.5032	0.5019	0.5044
	6:4	0.5020	0.4855	0.5184
	5:5	0.4999	0.4890	0.5108

TABLE IV
STEGANALYSIS EVALUATION OF LIN ET AL.'S METHODS FOR STEGANOGRAPHIC ALGORITHMS (SE)

	Ratio (TRN:VAL:TST)	P_{E}
Lin et al.'s [59]	5:1:4	0.5006
	6:1:3	0.5033
	7:1:2	0.5165
	8:1:1	0.5115

The results are presented in Table IV, which reveals that even under such a large scale, the detection error rates are still close to 50%. The false alarm rate and missed detection rate are also consistently near 50%. These results suggest that it is challenging to distinguish between the distribution of speech with secret information and the original speech.

We also intentionally adjusted the ratio between the training and testing datasets to evaluate its impact on the security analysis. Notably, we observed that our method consistently maintained its indistinguishability between stego and cover, even as the dataset's balance underwent irregular changes. This distinguishes our approach from many other empirically secure steganographic techniques. Under CHA attacks, our method remains robust regardless of the dataset's unpredictable shifts in balance.

VII. CONCLUSION

In this paper, we propose a practical and complete public-key steganography construction based on elliptic curve cryptography. First, we construct an IND$-CPA secure public-key encryption construction using the Elligator2 technique. Using this scheme, we then construct a public-key steganography system based on reversible mapping and sampling. Next, we provide an instance of public-key steganography using WaveGlow. The proposed scheme is proven to be resistant to CHA and verified through NIST pseudorandomness tests and steganalysis experiments. Additionally, we propose a single-round asynchronous algorithm for steganographic key exchange. Finally, message integrity can be achieved by appending Message Authentication Codes (MACs) to the message ciphertext. As for identity authentication, it can be accomplished by using a public-private key pair duality to perform a pseudorandom signature.

In terms of the limitations of our work, firstly, it's worth noting that CHA attacks are essentially passive. We did not

consider scenarios where attackers actively attempt to impersonate or maliciously tamper with data to probe steganography users. This omission stems from the fact that, in practice, the number of steganographers is typically quite low compared to regular users, making such attacks costly. However, this may not hold in other contexts.

Additionally, regarding the issue of how public keys are obtained, our proposed approach relies on utilizing existing legitimate public keys. However, this may not meet the parameter requirements of Elligator 2. Nevertheless, in more recent Elligator variations like Elligator Squared [60] or Elligator Swift [61], the parameter requirements have been significantly relaxed and can be updated for practical use.

Furthermore, we aspire to extend more communication protocols into the realm of steganography, including group signatures and ring signatures in covert application scenarios. In the future, we may be able to establish a parallel world in the context of large-scale applications like AIGC, where we have communication tools similar to those in the real world but remain undetectable to external observers.

APPENDIX

Four theorems are listed below to prove that the functions $(\phi\ \ \psi)$ create a bijective function relationship between approximately half of the elliptic curve points and uniform random strings.

Theorem 3: $R = F_p\backslash\{0\}$, *if* $p \bmod 4 = 1$ *and* $\chi(A^2 - 4B) = -1$.

As per THEOREM 1, the decode function's extended domain is F_p when the curve meets specific requirements. This finding indicates that we can apply the decode function to all elements within $F_p\backslash 0$.

Extend domain $\hat{R} \triangleq R \cup \{0\} = F_p$.

Theorem 4: $\phi(\hat{R}) = E_R(F_p)$, *if* $p \bmod 4 = 1$ *and* $\chi(A^2 - 4B) = -1$.

THEOREM 2 demonstrates that the Encode Function's domain is identical to that of the Decode Function. During the proof of this proposition (refer to the appendix), we have established that $\phi(\psi((x, y)) = (x, y)$.

Theorem 5: $\phi(r) = \phi(r') \Leftrightarrow r' \in \{r, -r\}$. *where* $r, r' \in \hat{R}$.

As per Theorem 3, any nonzero element $(x, y) \in \phi(\hat{R})$ corresponds precisely to two preimages in F_p. Additionally, the two preimages satisfy $r = -r'$, and $(0, 0)$ has only one preimage $r = 0$.

Theorem 6: $\|E_R(F_p)\| = \frac{\|R\|}{2} + 1 = \frac{\|F_p\|+1}{2}$, $\|\cdot\|$ *is the number of elements in the set.*

Theorem 4 demonstrates the existence of $\frac{\|F_p\|+1}{2} = \frac{p+1}{2}$ encodable points on an elliptic curve. These points form $E_R(F_p)$. Defining $\hat{R} = 0, 1, 2, 3, \ldots, \frac{p-1}{2}$ creates a bijective relationship between $\hat{R}$ and $E_R(F_p)$. Therefore, any arbitrary string from $\hat{R}$ can be decoded as an elliptic curve point in $E_R(F_p)$ by ϕ, and any elliptic curve point in $\hat{R}$ can be encoded by ψ as a string belonging to $\hat{R}$.

A. Proof of THEOREM 3

Proof: The curve should satisfy $AB(A^2 - 4B) \neq 0$, so $A \neq 0,\ B \neq 0,\ A^2 - 4B \neq 0$. $\forall\ r \in F_p$ and $r \neq 0$, By the definition of R, $s = ur^2$, $\chi(u) = 1$, therefore, $\chi(s) \in \{0, 1\}$. Since $p\ mod\ 4 = 1$, $\chi(-1) = \chi(1) = 1$, $s \neq \pm 1$.

Then we assume that $A^2 s = B(1 + s)^2$,

$$\begin{cases} A^2 s = B(1+s)^2, \\ (A^2 - 4B)s = B(1-s)^2. \end{cases}$$

Thus, $(A^2 - 4B)A^2 s^2 = B^2(1+s)^2(1-s)^2$, $\chi(s) \in \{0, -1\}$, so $\chi((A^2-4B)A^2s^2) = \chi(B^2(1+s)^2(1-s)^2) = 1$, contrary to $\chi(A^2-4B) = -1$. Consequently, $A^2 s \neq B(1+s)^2$, $r \in R$. Since r is chosen arbitrarily, $F_p\backslash\{0\} \subseteq R$. By definition of R, we have $R \subseteq F_p\backslash\{0\}$. In conclusion, $R = F_q\backslash\{0\}$. □

B. Remark

Here, we will present some proofs of completeness of definitions, as well as explanations about the properties of R and ϕ.

- v is defined to be complete and $v \neq 0$. This follows from $1 + ur^2 \neq 0$ and $A \neq 0$.
- $v^3 + Av^2 + Bv \neq 0$, i.e. $\epsilon \neq 0$. Assuming that $v^3 + Av^2 + Bv \neq 0$ holds, we have $v^2 + Av = -B$. By the definition $v = -A/(1 + ur^2)$, we have $v + vur^2 = -A$. Therefore, $v^2 + Av = -v^2 ur^2 = -B$, which contradicts $A^2 ur^2 \neq B(1 + ur^2)^2$.
- $\chi(x^3 + Ax^2 + Bx) = 1$, and thus the definition of y is complete.
- $x \neq 0$ and $y \neq 0$, it follows from $\chi(x^3 + Ax^2 + Bx) = 1 \neq 0$ and $\epsilon \neq 0$.
- $x \neq -A$. If $\phi(v^3 + Av^2 + Bv) = 1 = \epsilon$, then $x = v$ since $v + vur^2 = -A$ and $vur^2 \neq 0$ (because $v, u, r \neq 0$); hence, $x \neq -A$. If $\phi(v^3 + Av^2 + Bv) = -1 = \epsilon$, then $x = -v - A \neq 0$.
- $\phi(r) = \phi(-r)$. It follows from $v = \frac{-A}{1+ur^2}$ and the single-valuedness of the square root function with respect to r^2.

C. Proof of THEOREM 4

Proof: ($\Rightarrow$) Proof that $\phi(\hat{R}) \subseteq E_R(F_q)$.

Obviously, $\phi(0) = (0, 0) \in E_R(F_q)$. Now consider the case $r \neq 0$. Let $r \in \hat{R}\backslash 0 = R$ be arbitrary, and suppose $\phi(r) = (x, y)$. By Remark, we have $x \neq -A$ and $y \neq 0$. Additionally, from $x(x+A) = v(v+A)$, we obtain $-ux(x+A) = -uv(v+A) = u^2v^2r^2$. Note that $u, v, r \neq 0$, so $\chi(-ux(x + A)) = \chi(u^2v^2r^2) = 1$. Therefore, $\phi(r) = (x, y) \in E_R(F_q)$.

Thus, we have shown that $\phi(\hat{R}) \subseteq E_R(F_q)$.

($\Leftarrow$) Prove that $\phi(\hat{R}) \supseteq E_R(F_q)$.

We will show that for any $(x, y) \in E_R(F_q)$, there exists $\bar{r} \in \hat{R}$ such that $\phi(\bar{r}) = (x, y)$. If $(x, y) = (0, 0)$, we can choose $\bar{r} = 0$, and $\phi(\bar{r}) = (0, 0)$ holds.

Now consider the case where $y \neq 0$. By the definition of $E_R(F_q) \supseteq E(F_q)$, we have $y^2 = x^3 + Ax^2 + Bx$ where $A \neq 0, B \neq 0$, and $AB(A^2 - 4B) \neq 0$. Therefore, we have $x \neq 0$ and $x^2 + Ax + B \neq 0$. Additionally, $\chi(x^3 + Ax^2 + Bx) = 1$. Since $x \neq -A$, we have $(x + A) \neq 0$. Let u be a fixed parameter with $u \neq 0$ and $\chi(u) = -1$.

Therefore, both $\frac{-x}{(x+A)u}$ and $\frac{-(x+A)}{ux}$ are well defined, and $\chi(\frac{-x}{(x+A)u}) = \chi(\frac{-(x+A)}{ux}) = \chi(-ux(x + A)) = 1$.

Define $\bar{r}$ as follows: $\begin{cases} \sqrt{\frac{-x}{(x+A)u}}, & if\ y \in \sqrt{F_q^2} \\ \sqrt{\frac{-(x+A)}{ux}}, & if\ y \notin \sqrt{F_q^2} \end{cases}$ $\bar{r}$ is well defined. By the definition of $\bar{r}$, $\bar{r} \neq 0$.

We now prove that $\bar{r} \in R \subseteq \hat{R}$, i.e. $1+u\bar{r}^2 \neq 0$, $A^2u\bar{r}^2 \neq B(1+u\bar{r}^2)^2$. Our idea is to use ϕ to try to decode $\bar{r}$ and obtain a series of temporary variables to assist in our proof.

All calculations below are assumed to be complete.

(1) If $y \in \sqrt{F_q^2}$ and $y = \sqrt{x^3+Ax^2+Bx}$, then $\bar{r}^2 = \frac{-x}{(x+A)u}$, $1+u\bar{r}^2 = \frac{A}{A+x} \neq 0$. Let $\bar{v} = \frac{-A}{1+u\bar{r}^2} = -(A+x)$, $\chi(\bar{v}) = \chi(-x-A) = \chi(ux) = -\chi(x)$, $\bar{v}^2 + A\bar{v} + B = (A+x)^2 - A(A+x) + B = x^2 + Ax + B$. Let $\bar{\epsilon} = \chi(\bar{v}^3 + A\bar{v}^2 + B\bar{v}) = \chi(\bar{v})\chi(\bar{v}^2+A\bar{v}+B) = -\chi(x)\chi(x^2+Ax+B) = -\chi(x^3+Ax^2+Bx) = -1$. Let $\bar{x} = \bar{\epsilon}\bar{v} - \frac{(1-\bar{\epsilon})A}{2} = -\bar{v} - A = -(-(A+x)) - A = x$. Let $\bar{y} = -\bar{\epsilon}\sqrt{\bar{x}^3 + A\bar{x}^2 + B\bar{x}} = -\bar{\epsilon}\sqrt{x^3+Ax^2+Bx} = y$.

(2) If $y \notin \sqrt{F_q^2}$ and $y = -\sqrt{x^3+Ax^2+Bx}$, $\bar{r}^2 = \frac{-(x+A)}{ux}$, $1+u\bar{r}^2 = \frac{-A}{x} \neq 0$, then let $\bar{v} = \frac{-A}{1+u\bar{r}^2} = x$, $\bar{\epsilon} = \chi(\bar{v}^3+A\bar{v}^2+B\bar{v}) = \chi(x^3+Ax^2+Bx) = 1$, $\bar{x} = \bar{\epsilon}\bar{v} - \frac{(1-\bar{\epsilon})A}{2} = \bar{v} = x$, and $\bar{y} = -\bar{\epsilon}\sqrt{\bar{x}^3 + A\bar{x}^2 + B\bar{x}} = -\sqrt{x^3+Ax^2+Bx} = y$.

Both (1) and (2) have $1+u\bar{r}^2 \neq 0$ and $\bar{v}^2 + A\bar{v} + B = x^2+Ax+B$. Substituting $\bar{v}$, we have $\bar{v}^2+A\bar{v}+B = \frac{A^2}{(1+u\bar{r}^2)^2} - \frac{A^2}{1+u\bar{r}^2} + B = \frac{B(1+u\bar{r}^2)^2 - A^2u\bar{r}^2}{(1+u\bar{r}^2)^2}$. Therefore, we have $-A^2u\bar{r}^2 + B(1+u\bar{r}^2)^2 = (1+u\bar{r}^2)^2(\bar{v}^2 + A\bar{v} + B) \neq 0$, noting that $(1+u\bar{r}^2) \neq 0$ and $x^2 + Ax + B \neq 0$.

Overall, we have $\begin{cases} \bar{r} \neq 0, \\ 1+u\bar{r}^2 \neq 0, \\ A^2u\bar{r}^2 \neq B(1+u\bar{r}^2)^2. \end{cases} \Rightarrow \bar{r} \in R.$

Therefore, for any $(x, y) \in E_R(F_q)$, there exists $\bar{r} \in \hat{R}$ such that $\phi(\bar{r}) = (x, y)$, which implies that $\phi(\hat{R}) \supseteq E_R(F_q)$.

Combining the "if" and "only if" statements, we have $\phi(\hat{R}) = E_R(F_q)$. □

D. Proof of THEOREM 5

($\Rightarrow$) When $r = 0$, it is easy to prove that $\phi(0) = (0, 0)$ and $\phi(\neq 0) = (x, y) \neq (0, 0)$ (by Remark), so there must exist $r' = 0$. For the case of $r, r' \neq 0$, we have $(x', y') = \phi(r') = \phi(r) = (x, y)$. Since $y = -\epsilon\sqrt{x^3+Ax^2+Bx} = y' = -\epsilon'\sqrt{x'^3 + Ax'^2 + Bx'}$, we have $\epsilon = \epsilon' \neq 0$ due to $x' = x$ and $x^3 + Ax^2 + Bx \neq 0$, and $v = v'$ due to $x = \epsilon v - \frac{(1-\epsilon)A}{2} = x' = \epsilon' v' - \frac{(1-\epsilon')A}{2}$. By $v = \frac{-A}{1+ur^2} = v' = \frac{-A}{1+ur'^2}$ (where $A, u \neq 0$), we have $r^2 = r'^2$, thus $r' \in r, -r$.

($\Leftarrow$) The fact that $\phi(r) = \phi(-r)$ has already been proven in Remark.

E. Proof of THEOREM 6

Combining THEOREM 3, THEOREM 4 and THEOREM 5, ϕ and ψ are bijection functions. Therefore, $\|E_R(F_p)\| = \frac{\|R\|}{2} + 1 = \frac{\|F_p\|+1}{2}$.

REFERENCES

[1] R. J. Anderson and F. A. P. Petitcolas, "On the limits of steganography," *IEEE J. Sel. Areas Commun.*, vol. 16, no. 4, pp. 474–481, May 1998.

[2] L. M. Marvel, C. G. Boncelet, and C. T. Retter, "Spread spectrum image steganography," *IEEE Trans. Image Process.*, vol. 8, no. 8, pp. 1075–1083, Aug. 1999.

[3] I. Cox, M. Miller, J. Bloom, J. Fridrich, and T. Kalker, *Digital Watermarking and Steganography*. Burlington, MA, USA: Morgan Kaufmann, 2007.

[4] D. Wang and G. Mark, "Internet censorship in China: Examining user awareness and attitudes," *ACM Trans. Comput.-Hum. Interact.*, vol. 22, no. 6, pp. 1–22, Dec. 2015.

[5] R. Ramesh et al., "Decentralized control: A case study of Russia," in *Proc. Netw. Distrib. Syst. Secur. Symp. (NDSS)*, 2020.

[6] P. Gill, M. Crete-Nishihata, J. Dalek, S. Goldberg, A. Senft, and G. Wiseman, "Characterizing web censorship worldwide: Another look at the OpenNet initiative data," *ACM Trans. Web (TWEB)*, vol. 9, no. 1, pp. 1–29, 2015.

[7] G. J. Simmons, "The prisoners' problem and the subliminal channel," in *Proc. Adv. Cryptology, Crypto*, vol. 83, 1984, pp. 51–67.

[8] J. Fridrich and J. Kodovsky, "Rich models for steganalysis of digital images," *IEEE Trans. Inf. Forensics Security*, vol. 7, pp. 868–882, 2012.

[9] J. Mielikainen, "LSB matching revisited," *IEEE Signal Process. Lett.*, vol. 13, no. 5, pp. 285–287, May 2006.

[10] X. Zhang and S. Wang, "Efficient steganographic embedding by exploiting modification direction," *IEEE Commun. Lett.*, vol. 10, no. 11, pp. 781–783, Nov. 2006.

[11] J. Fridrich, M. Goljan, and R. Du, "Detecting LSB steganography in color, and gray-scale images," *IEEE MultimediaMag.*, vol. 8, no. 4, pp. 22–28, Apr. 2001.

[12] B. Feng, W. Lu, and W. Sun, "Secure binary image steganography based on minimizing the distortion on the texture," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 2, pp. 243–255, Feb. 2015.

[13] T. Pevnỳ, T. Filler, and P. Bas, "Using high-dimensional image models to perform highly undetectable steganography," in *Proc. Inf. Hiding,12th Int. Conf.*, Calgary, AB, Canada. Berlin, Germany: Springer, 2010, pp. 161–177.

[14] W. Li, W. Zhang, L. Li, H. Zhou, and N. Yu, "Designing near-optimal steganographic codes in practice based on polar codes," *IEEE Trans. Commun.*, vol. 68, no. 7, pp. 3948–3962, Jul. 2020.

[15] W. Tang, S. Tan, B. Li, and J. Huang, "Automatic steganographic distortion learning using a generative adversarial network," *IEEE Signal Process. Lett.*, vol. 24, no. 10, pp. 1547–1551, Oct. 2017.

[16] J. Yang, D. Ruan, J. Huang, X. Kang, and Y.-Q. Shi, "An embedding cost learning framework using gan," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 839–851, 2019.

[17] X. Mo, S. Tan, B. Li, and J. Huang, "MCTSteg: A Monte Carlo tree search-based reinforcement learning framework for universal non-additive steganography," *IEEE Trans. Inf. Forensics Security*, vol. 16, pp. 4306–4320, 2021.

[18] X. Mo, S. Tan, W. Tang, B. Li, and J. Huang, "ReLOAD: Using reinforcement learning to optimize asymmetric distortion for additive steganography," *IEEE Trans. Inf. Forensics Security*, vol. 18, pp. 1524–1538, 2023.

[19] J. Ye, J. Ni, and Y. Yi, "Deep learning hierarchical representations for image steganalysis," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 11, pp. 2545–2557, Jun. 2017.

[20] M. Boroumand, M. Chen, and J. Fridrich, "Deep residual network for steganalysis of digital images," *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 5, pp. 1181–1193, Jul. 2018.

[21] J. Yang, Z. Yang, J. Zou, H. Tu, and Y. Huang, "Linguistic steganalysis toward social network," *IEEE Trans. Inf. Forensics Security*, vol. 18, pp. 859–871, 2023.

[22] C. Cachin, "An information-theoretic model for steganography," in *Proc. Inf. Hiding, 2nd Int. Workshop (IH)*, Portland, OR, USA. Springer, Apr. 1998, pp. 306–318.

[23] N. J. Hopper, J. Langford, and L. Von Ahn, "Provably secure steganography," in *Proc. Adv. Cryptology-CRYPTO 22nd Annu. Int. Cryptol. Conf.*, Santa Barbara, CA, USA. Berlin, Germany: Springer, Aug. 2002, pp. 77–92.

[24] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 214–223.

[25] R. Jozefowicz, O. Vinyals, M. Schuster, N. Shazeer, and Y. Wu, "Exploring the limits of language modeling," 2016, *arXiv:1602.02410*.

[26] R. Prenger, R. Valle, and B. Catanzaro, "WaveGlow: A flow-based generative network for speech synthesis," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 3617–3621.

[27] K. Yang, K. Chen, W. Zhang, and N. Yu, "Provably secure generative steganography based on autoregressive model," in *Proc. Digital Forensics Watermarking, 17th Int. Workshop (IWDW)*, Jeju Island, South Korea. Cham, Switzerland: Springer, Oct. 2018, pp. 55–68.

[28] K. Chen, H. Zhou, H. Zhao, D. Chen, W. Zhang, and N. Yu, "Distribution-preserving steganography based on text-to-speech generative models," *IEEE Trans. Dependable Secure Comput.*, vol. 19, no. 5, pp. 3343–3356, Sep. 2022.

[29] Z. M. Ziegler, Y. Deng, and A. M. Rush, "Neural linguistic steganography," 2019, *arXiv:1909.01496*.

[30] S. Zhang, Z. Yang, J. Yang, and Y. Huang, "Provably secure generative linguistic steganography," 2021, *arXiv:2106.02011*.

[31] G. Kaptchuk, T. M. Jois, M. Green, and A. D. Rubin, "Meteor: Cryptographically secure steganography for realistic distributions," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, Nov. 2021, pp. 1529–1548.

[32] J. Ding, K. Chen, Y. Wang, N. Zhao, W. Zhang, and N. Yu, "Discop: Provably secure steganography in practice based on 'distribution copies,"' in *Proc. IEEE Symp. Secur. Privacy (SP)*, May 2023, pp. 2238–2255.

[33] C. S. de Witt, S. Sokota, J. Z. Kolter, J. Foerster, and M. Strohmeier, "Perfectly secure steganography using minimum entropy coupling," 2022, *arXiv:2210.14889*.

[34] L. Von Ahn and N. J. Hopper, "Public-key steganography," in *Proc. Adv. Cryptol.-EUROCRYPT Int. Conf. Theory Appl. Cryptograph. Techn.*, Interlaken, Switzerland. Berlin, Germany: Springer, May 2004, pp. 323–341.

[35] N. Hopper, "On steganographic chosen covertext security," in *Proc. Automata, Lang. Programming, 32nd Int. Colloq. (ICALP)*, Lisbon, Portugal. Berlin, Germany: Springer, Jul. 2005, pp. 311–323.

[36] M. Backes and C. Cachin, "Public-key steganography with active attacks," in *Proc. Theory Cryptography, 2nd Theory Cryptography Conf. (TCC)*, Cambridge, MA, USA. Berlin, Germany: Springer, Feb. 2005, pp. 210–226.

[37] D. J. Bernstein, M. Hamburg, A. Krasnova, and T. Lange, "Elligator: Elliptic-curve points indistinguishable from uniform random strings," in *Proc. ACM SIGSAC Conf. Comput. Commun. Sec.*, 2013, pp. 967–980.

[38] D. Boneh, "The decision Diffie–Hellman problem," in *Proc. Algorithmic Number Theory, 3rd Int. Symp. ANTS-III*, Portland, OR, USA. Berlin, Germany: Springer, Jun. 2006, pp. 48–63.

[39] D. J. Bernstein, "Curve25519: New Diffie–Hellman speed records," in *Proc. Public Key Cryptography-PKC 9th Int. Conf. Theory Pract. Public-Key Cryptography*, New York, NY, USA. Berlin, Germany: Springer, Apr. 2006, pp. 207–228.

[40] A. Rukhin, J. Soto, J. Nechvatal, M. Smid, and E. Barker, "A statistical test suite for random and pseudorandom number generators for cryptographic applications," Booz-Allen Hamilton, McLean, VA, USA, Tech. Rep. NIST SP 800-22, 2001.

[41] W. Luo, H. Li, Q. Yan, R. Yang, and J. Huang, "Improved audio steganalytic feature and its applications in audio forensics," *ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 14, no. 2, pp. 1–14, May 2018.

[42] Q. Liu, A. H. Sung, and M. Qiao, "Temporal derivative-based spectrum and mel-cepstrum audio steganalysis," *IEEE Trans. Inf. Forensics Security*, vol. 4, no. 3, pp. 359–368, Sep. 2009.

[43] Q. Liu, A. H. Sung, and M. Qiao, "Derivative-based audio steganalysis," *ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 7, no. 3, pp. 1–19, Aug. 2011.

[44] C. E. Shannon, "Communication theory of secrecy systems," *Bell Syst. Tech. J.*, vol. 28, no. 4, pp. 656–715, Oct. 1949.

[45] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Commun. ACM*, vol. 21, no. 2, pp. 120–126, Feb. 1978.

[46] T. Elgamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Trans. Inf. Theory*, vol. IT-31, no. 4, pp. 469–472, Jul. 1985.

[47] N. Koblitz, "Elliptic curve cryptosystems," *Math. Comput.*, vol. 48, no. 177, pp. 203–209, 1987.

[48] S. Katzenbeisser and F. A. Petitcolas, "Defining security in steganographic systems," *Proc. SPIE*, vol. 4675, pp. 50–56, Apr. 2002.

[49] T. Van Le, "Efficient provably secure public key steganography," *Cryptol. ePrint Arch.*, 2003.

[50] I. Goodfellow, "Generative adversarial networks," *Commun. ACM*, vol. 63, no. 11, pp. 139–144, 2020.

[51] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," 2013, *arXiv:1312.6114*.

[52] *Gpt-4 Technical Report*, OpenAI, San Francisco, CA, USA, 2023.

[53] D. Holz et al. (2022). *Midjourney*. Artificial Intelligence Platform. Accessed: Nov. 1, 2022. [Online]. Available: https://www.midjourney.com/

[54] V. Liu, H. Qiao, and L. Chilton, "Opal: Multimodal image generation for news illustration," in *Proc. 35th Annu. ACM Symp. User Interface Softw. Technol.*, Oct. 2022, pp. 1–17.

[55] C. Wang, B. Chen, Z. Duan, W. Chen, H. Zhang, and M. Zhou, "Generative text convolutional neural network for hierarchical document representation learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 4, pp. 4586–4604, Apr. 2023.

[56] J. Liu, C. Li, Y. Ren, F. Chen, and Z. Zhao, "DiffSinger: Singing voice synthesis via shallow diffusion mechanism," in *Proc. AAAI Conf. Artif. Intell.*, 2022, vol. 36, no. 10, pp. 11020–11028.

[57] J. H. Silverman, *The Arithmetic of Elliptic Curves*, vol. 106. New York, NY, USA: Springer, 2009.

[58] J. Behrmann, W. Grathwohl, R. T. Chen, D. Duvenaud, and J.-H. Jacobsen, "Invertible residual networks," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 573–582.

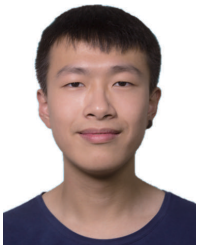
[59] Y. Lin, R. Wang, D. Yan, L. Dong, and X. Zhang, "Audio steganalysis with improved convolutional neural network," in *Proc. ACM Workshop Inf. Hiding Multimedia Secur.*, Jul. 2019, pp. 210–215.

[60] M. Tibouchi, "Elligator squared: Uniform points on elliptic curves of prime order as uniform random strings," in *Proc. Int. Conf. Financial Cryptography Data Secur.* Berlin, Germany: Springer, 2014, pp. 139–156.

[61] J. Chávez-Saab, F. Rodríguez-Henríquez, and M. Tibouchi, "Swiftec: Shallue-van de woestijne indifferentiable function to elliptic curves: Faster indifferentiable hashing to elliptic curves," in *Proc. Int. Conf. Theory Appl. Cryptol. Inf. Secur.* Cham, Switzerland: Springer, 2022, pp. 63–92.

Xin Zhang received the B.S. degree from the University of Science and Technology of China (USTC), Hefei, China, in 2022, where he is currently pursuing the master's degree in engineering. His research interests include information hiding, applied cryptography, and deep learning.

Kejiang Chen (Member, IEEE) received the B.S. degree from Shanghai University (SHU) in 2015 and the Ph.D. degree from the University of Science and Technology of China (USTC) in 2020. He is currently an Associate Research Fellow with the USTC. His research interests include information hiding, image processing, and deep learning.



Jinyang Ding (Member, IEEE) received the B.S. degree from the China University of Mining and Technology in 2020. He is currently pursuing the master's degree with the University of Science and Technology of China. His research interests include information hiding, privacy protection, and deep learning.

Yuqi Yang is currently pursuing the bachelor's degree with the University of Science and Technology of China. His research interests include information hiding, deep learning, and quantum networks.

Weiming Zhang received the M.S. and Ph.D. degrees from the Zhengzhou Information Science and Technology Institute, China, in 2002 and 2005, respectively. He is currently a Professor with the School of Information Science and Technology, University of Science and Technology of China. His research interests include information hiding and multimedia security.

Nenghai Yu received the B.S. degree from the Nanjing University of Posts and Telecommunications in 1987, the M.E. degree from Tsinghua University in 1992, and the Ph.D. degree from the University of Science and Technology of China (USTC) in 2004. He is currently a Professor with USTC. His research interests include multimedia security, multimedia information retrieval, video processing, and information hiding.