Traceable Alias Protocol based on Implicit Certificate

Wu-Chuan Yang[1,2], Lien-Yuan Ting[1], Kai Chain[2*]

[1]Department of Information Engineering, I-Shou University

[2*]Department of Intelligent Network Technology, I-Shou University

*Corresponding Author: kaichain@isu.edu.tw

DOI: https://doi.org/10.30211/JIC.202402.006

ABSTRACT

With the rapid development of the Internet, its anonymity and freedom have fostered the diversification of online applications in services and industry development, allowing users to access a wealth of convenient information easily. However, this has also led to numerous improper online behaviors, including crimes committed through hiding and falsifying identities. Implementing an online real-name system to prevent the hiding and falsification of identities is an important aspect of safety and the verification of true identities. Yet, this system faces controversies regarding the culture of online anonymity and individual privacy rights. To balance safety and convenience, this study is based on Rabadi's concept of implicit certificates, proposing a "Traceable Anonymity Certificate Application Protocol." By integrating certificate and digital signature mechanisms, this protocol aims to find a middle ground between real-name and anonymity systems. Besides standardizing identity verification behaviors and raising the threshold for identity impersonation, it also allows for the concealment of users' real identities to achieve anonymity. In case of disputes, users' identity information can be verified through specific methods, such as comparing the digital signature with the one stored in the certificate authority's database or using a secure hash function to verify the certificate's integrity.

Keywords: Implicit certificates, Identity tracing, Elliptic curve cryptography

1. Introduction

While the Internet brings convenience to people's lives, it also comes with many inappropriate online behaviors and threats to information security, including criminal activities conducted through identity concealment or forgery [1]. Regarding the overview of computer network crimes in our country, over the past five years, the incidence of online crimes has seen a 4.41% increase in cases of "fraud," making it the most prevalent, while "intellectual property infringement" has decreased by 7.57%, showing the most improvement. It can be observed that most online crimes take advantage of the anonymity and freedom provided by the Internet, making online crimes more widespread and rapid compared to conventional crimes. While using many convenient functions, it is also necessary

to prevent attackers' actions to protect sensitive information from being stolen or tampered with. Servers must be able to verify users' identities without interference, ensuring user identity is confirmed without exposing all real data. Most anonymization technologies are designed to protect privacy. For example, proxy anonymization uses methods such as forwarding to have the user's packets sent by the proxy server. However, malicious users exploit proxy anonymization to cover their tracks and commit crimes. Using real-name authentication for user identity verification can prevent anonymity, but it raises privacy concerns. It is not suitable for certain scenarios. For example, social networking sites are communication platforms where every user's real information doesn't have to be exposed on a public platform.

Regarding policy, personal privacy on the Internet is highly valued, and various countries have established related regulations. For example, domestically, the implementation of personal data protection is governed by the "Enforcement Rules of the Computer-Processed Personal Data Protection Act," which has been amended and renamed as the "Enforcement Rules of the Personal Data Protection Act" [2]. "Anyone who, with the intent to unlawfully benefit themselves or a third party or to harm the interests of others, unlawfully alters, deletes, or otherwise manipulates personal data files, thereby impairing the accuracy of the personal data files and causing damage to others, shall be subject to imprisonment under the Criminal Code and fined in New Taiwan Dollars." Individuals, businesses, and government entities inadvertently violating this law will face significant compensation liabilities.

To effectively address the controversies surrounding Internet identity anonymization and real-name systems related to Internet freedom, this study proposes a theoretical "registered system" for users, which lies between real-name and anonymous systems. It involves the use of registered certificates to regulate identity authentication. Only specific entities and specific methods can unidirectionally track a user's real identity, while others cannot track the real identity of users. This system combines the advantages of both real-name and anonymous systems (shown in Figure 1).

To ensure that the recipient trusts the user's identity, we use a public key infrastructure (PKI). Users seek a trusted third party to act as a Certificate Authority (CA) responsible for issuing, revoking, and maintaining certificates. Generally, CA issues a certificate to the user. According to the X.509 standard, this certificate must include the user's identity, issuer's name, validity period, and algorithm, among other details [3, 4]. Our solution involves the user sending their own ID and other real information to the CA. The CA then generates a certificate based on the user's ID that does not include their real name; we refer to this as an Implicit Certificate. It contains a new public key and a "pseudonym." Anyone who receives the pseudonym certificate will only know the pseudonym, not the user's real identity. Only the CA can link the pseudonym to the user's true identity. In this way, the user can hide their real identity while the recipient can still trust the message's authenticity based on the pseudonym certificate.

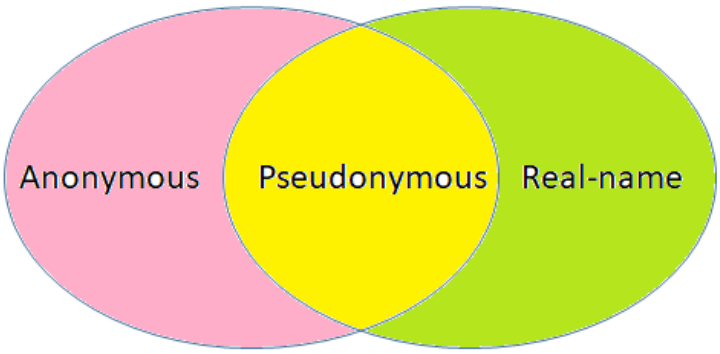

Figure 1. Traceable Pseudonymous Certificates Considering the Advantages of Both Real-name and Anonymous Systems (Source: By authors)

This article collects data on related Internet behaviors, regulations, and existing Internet anonymization behaviors to serve as the basis for our proposed solution. Secondly, relevant knowledge and literature focus on discussing Internet anonymization behaviors and analyzing the feasibility of the pseudonym solution. Finally, based on the concept of implicit certificates, using certificates and the Elliptic Curve, the ElGamal algorithm achieves the goal of the pseudonym system. This approach raises the threshold for identity forgery, prevents Internet anonymization, and allows for user identity verification without exposing all real data.

2. Literature Review

To implement pseudonymous certificates, we first introduce current certificate and cryptographic technologies, including Public Key Infrastructure (PKI), Elliptic Curve ElGamal Digital Signatures, and Bilinear Pairing.

2.1 Public Key Infrastructure (PKI)

Many Certificate Authorities (CA) require the verification of real identities when applying for certificates. Therefore, using certificates is a crucial step in the real-name system on the Internet. For example, in the case of online tax filing involving user rights, it is necessary to comply with real-name requirements on the Internet. Certificates link the user's personal identity with their public key, ensuring that each CA user's identity is unique, thereby enabling the identification of individual identities. Public Key Infrastructure (PKI) includes details related to the issuance of certificates, such as regulations, Certificate Authorities (CA), and relevant technical components. Certificates are currently considered an effective solution for insecure network environments, particularly in public settings where users do not know each other. Therefore, the security and authenticity of user keys are especially emphasized. To prove that a certain public key is indeed owned by a specific person or entity, a trusted third-party institution is used as a management center to verify the authenticity of the public key. Thus, CA is established to issue electronic certificates to validate the effectiveness of public keys. In general, the setup of PKI allows users to request authentication and use the public key information within the public key certificate to encrypt messages sent to others. Users utilize their

own private keys to decrypt messages.

PKI is also used by governments worldwide as the security infrastructure for e-government. For example, our country has established a hierarchical PKI based on the ITU-T X.509 standard domestically. This includes the Trust Anchor, the Government Root Certification Authority (GRCA), and subordinate CA set up by various government agencies. The GRCA issues CA certificates to the subordinate CA and their respective governing authorities. This system supports services such as online tax filing, electronic highway supervision, electronic invoicing, healthcare systems, and electronic medical records [5]. The current public key infrastructure framework of our government is shown in Figure 2.

Figure 2. Current Public Key Infrastructure Framework (https://grca.nat.gov.tw/index2.html)

PKI components include the Certificate Authority (CA), Registration Authority (RA), and repository. The CA is responsible for issuing public key certificates and assures the authenticity of the user's public key by signing the user's certificate with a digital signature, thereby preventing malicious actors from impersonating the user's public key. The RA establishes and verifies the applicant's identity and is responsible for processing certificate applications and data verification. The PKI requires a repository to store relevant certificate information, including certificates issued by the CA and the Certificate Revocation List (CRL). The repository provides the publication, inquiry, and download of CA certificates and revoked certificate lists and also offers certificate practice statements and related information.

2.2 Elliptic Curve ElGamal Digital Signature

In online communications, people want to attach a mechanism similar to a personal signature to certain important documents, and digital signatures provide a similar "electronic seal." The goal is to use digital signatures to verify that the message originates from a specific party.

The elliptic curve public key cryptography technology, proposed by Koblitz and Miller in 1985 at different conferences and journals [6, 7], can be applied not only in cryptographic encryption and

decryption, digital signatures, and key exchange but also in large integer factorization and primality testing. Elliptic curves are highly regarded in cryptography primarily because they offer better security at the same key length compared to basic modular arithmetic (including discrete logarithms and factorization) public key cryptosystems [17]. According to current evaluations of computational complexity, the security of a 160-bit elliptic curve cryptography (ECC) key is equivalent to that of a 1024-bit RSA key [8]. At the same level of security strength, ECC keys are shorter, and processing speeds are faster than RSA and DSA. This means that each key bit of ECC provides much more security than other public key cryptosystems, making it suitable for use in IC cards or devices with limited memory (shown in Table 1).

Table 1. NIST Recommended Key Sizes

Security Level	RSA Key Length	ECC Key Length
2^{80}	1024	160~223
2^{112}	2048	224~255
2^{128}	3072	256~383
2^{192}	7680	384~511
2^{256}	15360	521+

Source: By authors.

Public key cryptosystems based on the elliptic curve discrete logarithm problem must specify the curve parameter values:

1. Define the elliptic curve to be computed as $= E(q; a, b)$, $P = (x_1, y_1)$ and $Q = (x_2, y_2)$ are any two points on the curve, then:
   - Select a prime field $\mathrm{GF}(p)$, and let q be a prime number p; the curve is defined as $E: y^2 = x^3 + ax + b\ (mod\ p)$;
   - Select a binary field $\mathrm{GF}(2^m)$, and let q be an irreducible polynomial of degree m. The curve is defined as $E: y^2 + xy = x^3 + ax^b + b/F_{2^m}$.
2. Compute $g = \#E(F_q)$
3. Select a point P on $E(F_q)$ such that:
   - $n = \mathrm{ord}(P)$ has a large prime factor.
   - $h = {\#E(F_q)}/{\mathrm{ord}(P)}$ is very small.
4. The curve parameters $(E/F_q, P)$ represent $(q, a, b, g, x(P), y(P), ord(P), h)$.

This system is based on the Elliptic Curve ElGamal digital signature. Alice converts the message m into a signature s and sends it to Bob, where m is an integer and $0 \le m \le n$.

- Curve parameter agreement $({\mathrm{E}}/{F_q}, P)$
- Key Generation

Alice randomly selects an integer a such that $gcd(a, g) = 1$, and computes $P_a = [a]P$. The public key is ($^{E}/_{F_q}$, P, P_a), and the private key is a.

● Digital Signature

1. Alice randomly selects an integer *k* such that gcd (*k*, *g*)=1.
2. Compute $R = [k]P$.
3. Compute $s^* = k^{-1}(m - ax(R)) \bmod n$, where $x(R)$ is the x-coordinate of point *R*.
4. Send the signature $s = (m, R, s^*)$ to Bob.

● Verification

1. Bob receives $s = (m, R, s^*)$ and obtains Alice's public key ($^{E}/_{F_q}$, P, P_a).
2. Compute $V_1 = [x(R)]P_a + [s^*]R$, $V_2 = [m]P$.
3. If $V_1 = V_2$, accept; otherwise, reject.

The most important characteristics of digital signatures are non-repudiation and non-forgery. Only the person who possesses the private key can create a valid digital signature. An attacker attempting to forge a signature must solve the Elliptic Curve Discrete Logarithm Problem (ECDLP), which is computationally infeasible to solve within a reasonable amount of time. The ElGamal digital signature uses the private key and incorporates a random number in each signing process [16]. Therefore, even if the same signer signs the same plaintext, different signatures can still be produced.

2.3 Bilinear Pairing

The bilinear pairing functions, such as Weil Pairing and Tate Pairing [9-11, 18], are defined as linear mappings between two cyclic groups. The detailed explanation is as follows.

Let G_1 *and* G_2be additive groups, with generator point *P*, forming a cyclic multiplicative group G_T of size q. Define a function $e: G_1 \times G_2 \to G_T$ that maps points on the elliptic curve to the multiplicative group G_T. This bilinear pairing satisfies the following three properties:

1. Bilinear: Let P_1, $P_2 \in G_1$ and $Q \in G_2$, then: $e(P_1 + P_2, Q) = e(P_1, Q)e(P_2, Q)$, $e(aP, bQ) = e(P, Q)^{ab}$; where *a* and *b* belong to Z_q^*.
   When G_T is an Abelian ring and G_1, G_2 are group homeomorphisms, it satisfies: $e(aP, Q) = e(P, aQ)$.
2. Non-degenerate: There exist $P \in G_1$ and $Q \in G_2$ such that $e(P, Q) \neq 1$.
3. Computability: For any two points P *and* Q, there exists an efficient algorithm to compute $e(P, Q)$.

3. The Concept of Implicit Certificates

This section introduces the implicit certificate scheme proposed by Nader M. Rabadi, "Anonymous Group Implicit Certificate Scheme" [12]. In this scheme, Nader uses implicit certificates [13] and elliptic curve encryption to achieve user anonymity, identity verification, and data integrity. First, the system initialization is introduced, followed by the steps of key generation,

signing, and verification.

3.1 System Initialization

Generally, the CA issues a unique public key certificate to the recognized user. This certification binds to a unique public key. The certificate includes the user's public key, the user's identity, the name of the certificate issuer, and the encryption algorithm used by the CA to authenticate the certificate.

First, let *E* be an elliptic curve, *P* be a point on *E,* and the prime number *n* is the order of *P.* Let $c_A \in [1, n-1]$ be the CA's private key, and $C_A = c_A P$ be the public key. CA generates a private key $u_j \in [1, n-1]$ for a certain group *j* and a public key $U_j = u_j P$. CA also generates an implicit certificate I_j for this group. The implicit certificate I_j includes the group's minimal identity information, the CA's identity, and the certificate's validity period. Let *H* represent a secure hash function with an output length of $|H|$. CA computes $e_j = H(I_j || U_j)$.

Users register with the CA to apply for a certificate. CA assigns the users to a user group in the database. Let the identity of the user in group *j* be represented as *i*. The CA performs the following steps:

1. Generate the private keys $\{b_i, t_i\}$ and the public key $B_i = b_i P$ for user *i.*
2. Generate the signature $s_i = e_j u_j + b_i^{-1} c_A + t_i u_j \bmod n$ for user *i.*

Finally, the CA stores the user's private keys $\{b_i, t_i\}$, public key B_i, the CA's signature s_i, the CA's public key C_A, the group's public key U_j, and the implicit certificate of the group I_j.

3.2 Key Generation and Signing Procedures

Let *M* represent the user's message in the communication network, including a timestamp to protect against replay attacks. When user *i* is ready to broadcast *M*, the following steps are performed:

1. Compute $y = H(M)$.
2. Compute $\beta = y b_i s_i \bmod n$, $X = y b_i U_j$, and $Y = t_i X$. *X* is a base point on the curve.
3. Use the private key t_i to perform the digital signature algorithm on *y*. The Elliptic Curve DSA (Digital Signature Algorithm) is employed in this scheme. Assume *X* is a base point on $E(F_q)$ when signing *y*, then the signature $Sig_{t_i}(y)$ is generated.
4. Use the key β to sign the message $m = M||I_j||U_j||X||Y||Sig_{t_i}(y)$, generating the signature $Sig_\beta(m)$.
5. The user then broadcasts the message $m||Sig_\beta(m)$.

The verifier can use the known values e_j, y, the CA's public key C_A, and Y to construct the public key βP corresponding to the user's private key β. This is done as follows:

1. Compute the public key $\beta P = e_j X + Y + y C_A = Q$.
2. Use this public key Q to verify the signature.

The verification steps are as follows:

1. Compute $y = H(M)$ and $e_j = H(I_j||U_j)$.
2. Use the public key $Y = t_i X$ and the base point $X = y b_i U_j$ to verify the signature $Sig_{t_i}(y)$.
3. Construct the user's public key by computing $Q = e_j X + Y + y C_A$, and use Q to verify $Sig_\beta(m)$.

4. Traceable Anonymous Protocol Architecture

In the past, many scholars have proposed methods for user identity anonymity. For example, 2001, Rivest et al. proposed "How to leak a secret" [14]. By leveraging the signer identity uncertainty in ring signatures, they aimed to hide the user's true identity, thereby achieving anonymity. However, while these protocols satisfy user anonymity, they fail to enable user identity traceability. If a malicious user exploits anonymity for illegal activities, it becomes difficult to trace their identity.

User's identity information does not need to be exposed on public platforms such as general social networking sites and other open online platforms. Because our scheme ensures anonymity, users can protect their real information when posting on the site. However, if a user slanders or defames others on the site, the affected parties and the police can request the CA (Certificate Authority) to reveal the user's true identity.

Our proposed scheme involves three main roles: a trusted third-party CA, the user Alice, and the recipient Bob [15]. CA acts as a Trusted Third Party (TTP). If the CA behaves dishonestly or leaks user privacy by repeatedly signing the same user's key, the system cannot prevent this. The process includes system initialization, CA certificate issuance, user digital signing, and verification stages. The parameter definitions are shown in Table 2, and the system flowchart is shown in Figure 3. The detailed steps of the process are described in each subsection.

Table 2. System Architecture Parameter List

Parameter	Description
ID_A	User's identity
$Alias_A$	User's alias for login credentials
$sig(M)$	User's digital signature of their identity
P	A base point on the elliptic curve
P_{CA},r	CA's public and private keys
P_A,a	User's public and private keys
(E_A,V_A)	User's certificate

Source: By authors.

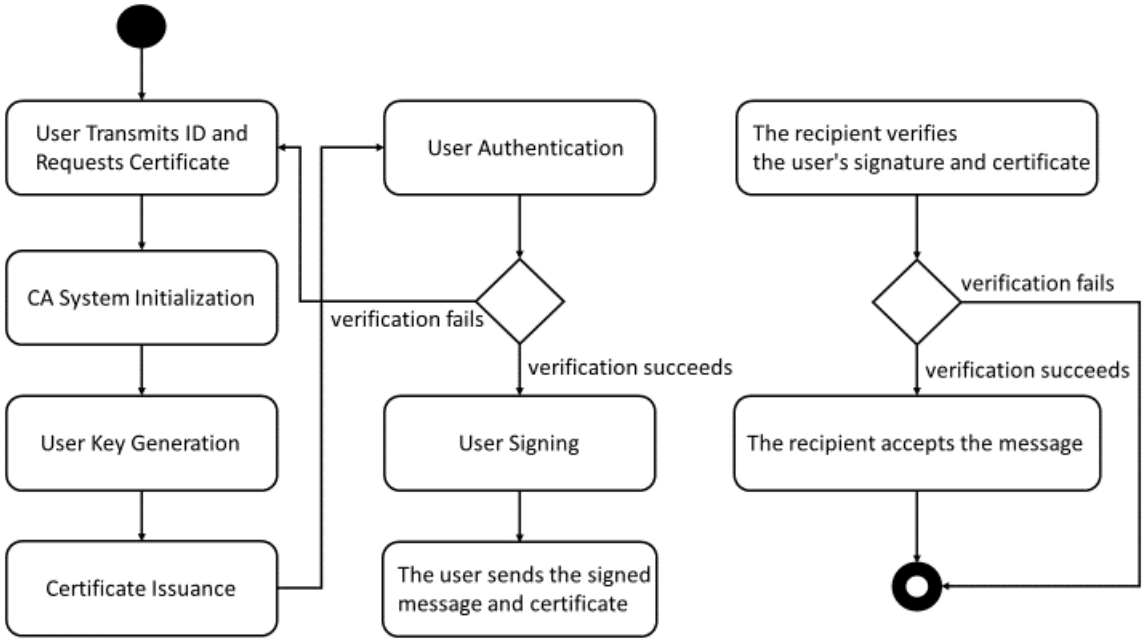

Figure 3. System Architecture Dynamic Diagram
(Source: By authors)

4.1 System Initialization

The user must first apply to the CA for a public/private key pair and a certificate. The user must possess a certificate before proceeding to the subsequent signing stages, as illustrated in Figure 4. Alice registers with the CA using her real information to apply for a named certificate and logs in with the username $Alias_A$. Alice digitally signs her identity information $M = (ID_A \,||Alias_A)$ using a digital signature algorithm such as ECDSA, calculates the signature $sig(M)$, and sends it to the CA. The CA often relies on hierarchical authentication and self-signed certificates stored in a public repository for identity verification, typically downloaded and stored in advance. In the initial stage, if one intends to conduct a man-in-the-middle attack between Alice and the CA, it often requires forging a website within the internal network and performing IP spoofing. This aspect necessitates strengthened internal security controls. Additionally, it requires forging the CA's self-signed certificate in advance, making the attack difficult to execute.

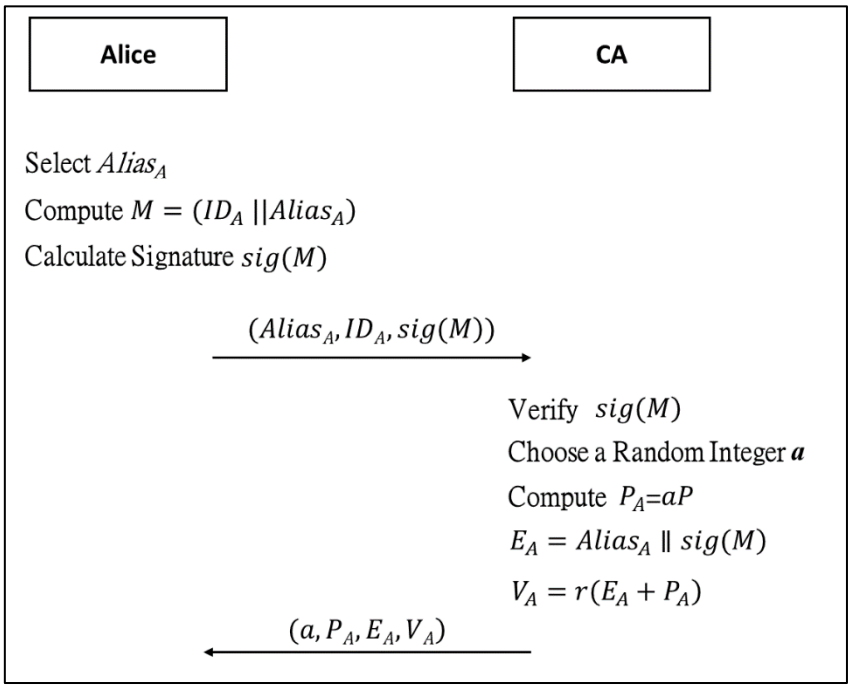

Figure 4. System Initialization (Source: By authors)

The CA obtains Alice's public key and verifies the signature $sig(M)$. If the verification is correct, the CA generates the private key a and the public key P_A=aP based on the information provided by Alice. The certificate components are $E_A = Alias_A \parallel sig(M)$, and $V_A = r(E_A + P_A)$. The CA sends the key pair $\{a,P_A\}$ and the certificate $\{E_A,V_A\}$ to Alice. The CA stores Alice's identity-related information and the signature $sig(M)$ onto a hardware device. After receiving the key pair (a,P_A) and the certificate (E_A,V_A), Alice verifies whether (E_A, V_A) is a valid certificate by checking if $e(V_A, P) = e((E_A + P_A)), P_{CA})$.

4.2 User Signing and Verification

User Signing Process: After Alice receives the certificate and key pair, she digitally signs the message m she intends to send, as illustrated in Figure 5.

Firstly, select a random integer k such that $\gcd(k, g) = 1$. Although the user's public key and private key are the same pair, each time a message is signed, a new random number is needed. The signature is computed using the following steps:

$$\begin{cases} R = [k]P \\ s^* = k^{-1}\big(m - a \times x(R)\big) \bmod n \end{cases}$$

Final Step: The user sends the digital signature s =(m,R,s^*) along with the certificate (E_i,V_i).

The verification process involves the recipient receiving the digital signature s and obtaining the user's public key $(E/F_q,\ P,P_i)$ to verify the certificate and the signature. First, verify $e(V_A, P) = e((E_A + P_A)), P_{CA})$, and check $V_1 = V_2$?

$$\begin{cases} V_1 = [x(R)]P_i + [s^*]R \\ V_2 = [m]P \end{cases}$$

If $V_1 = V_2$, the message will be accepted; otherwise, it will be rejected.

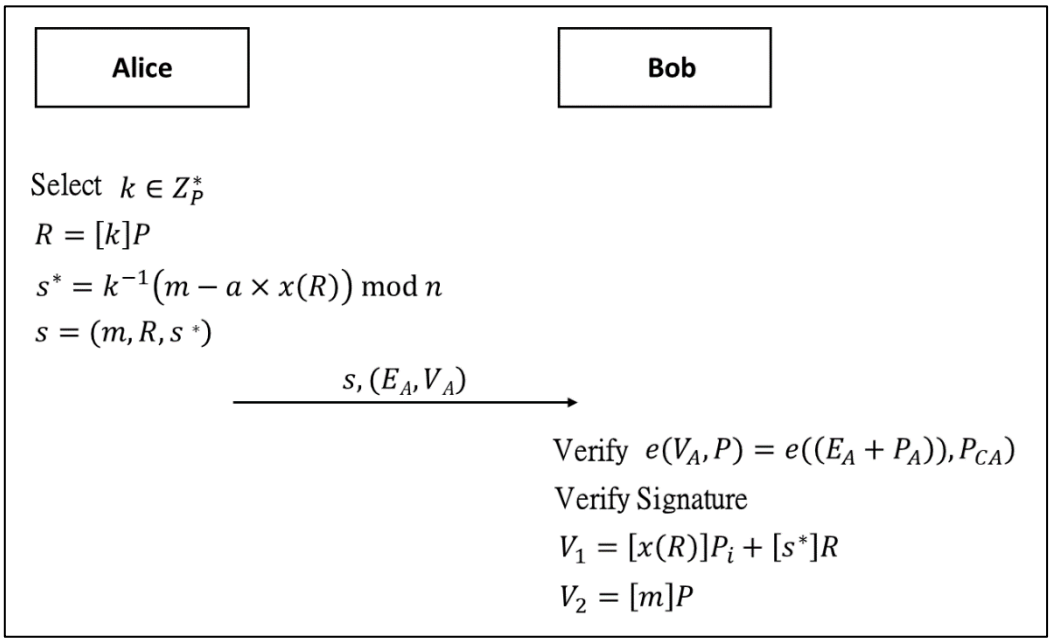

Figure 5. Signing and Verification Phases (Source: By authors)

Here are two examples of protocol applications:

Application 1: General Social Networking Sites and Other Open Online Platforms

Due to the implicit certificate providing user anonymity, users can hide their real information when posting on the website. However, if a user slanders or defames others on the site, the involved parties and the police can request the CA to reveal the user's true identity.

Application 2: Common Online Shopping or Other E-Commerce

Merchants primarily care about whether they can successfully receive payment or if the electronic cash is valid, and the user's real information is secondary. Therefore, consumers placing orders can use implicit certificates to maintain anonymity. If a consumer violates the contract, the merchant can request the CA to reveal the consumer's true identity for sanction.

5. Security Analysis

The security analysis of our proposed scheme includes correctness, anonymity, unforgeability, traceability, replay attack resistance, and non-repudiation. This paper does not discuss the performance aspect of implementing various bilinear pairing algorithms. The relevant analysis and comparison are as follows.

5.1 Correctness

Certificate Verification Proof: $e(V_A, P) = e((E_A + P_A), P_{CA})$

$$e(V_A, P) = e(r(E_A + P_A), P)$$
$$= e\big((E_A + P_A), rP\big)$$
$$= e\big((E_A + P_A), P_{CA}\big)$$

Signature Verification Proof: $V_1 = V_2$

$$V_1 = [x(R)]P_A + [s^*]R$$
$$= [x(R)][a]P + \left[k^{-1}\left(m - ax(R)\right)\right][k]P$$
$$= [a \times x(R) + m - a \times x(R)]P$$
$$= [m]P$$
$$= V_2$$

5.2 Anonymity

The CA encrypts the part containing Alice's real identity and embeds it into the certificate. Alice sends the message along with the certificate. Bob can verify the validity of the user's public key from the certificate by checking $e(V_A, P) = e((E_A + P_A), P_{CA})$. Bob cannot obtain any clues about the user's identity during the verification process. If Bob attempts to attack the certificate $E_A = Alias_A \parallel sig(M)$ to obtain the other party's identity, he will face the elliptic curve discrete logarithm problem. Additionally, except for the CA, others cannot obtain ID_A. An attacker cannot identify the real identity of a specific party from the certificate (E_A, V_A). Therefore, our system achieves anonymity.

5.3 Unforgeability

There is an attacker, Eve, who obtains Alice's public key P_A and forges Alice's signature on the message m. Eve must find two valid signatures, R and s^*, containing the message to forge Alice's signature. Eve selects R to compute s^*. She needs $[m]P = [x(R)]P_A + [s^*]R$. In other worlds, $s^* = ([m]P - [x(R)]P_A)R^{-1}$, where the variation of $R = [k]P$ depends on the randomly chosen nonce by the user. This implies that for Eve to forge the signature, she must solve the Elliptic Curve Discrete Logarithm Problem (ECDLP), which is extremely difficult. Our designed user-authenticated scheme is based on the assumption of the difficulty of solving the Elliptic Curve Discrete Logarithm Problem (ECDLP). (R, s^*) is a valid signature for m. Since Eve has no control over m, to forge the signature (m, R, s^*), she would need the user's random number k to do so. Therefore, the digital signature in this system remains secure. Thus, the designed scheme achieves unforgeability of signatures.

5.4 Traceability

Previous literature mostly mentions concealing user identities without considering the potential dangers of anonymity. The feature of a traceable authenticated protocol is that everyone can verify the validity of the certificate, but no one knows the true holder, except for the CA (Certificate Authority). When the holder engages in illegal activities and the court requires the CA to track the holder's true identity, the CA uses the content of E_A in the certificate to examine the user's signature $sig(M)$ to confirm Alice's true identity. The CA decrypts the certificate (E, V) containing the real information E_A to retrieve the user's real data. Thus, our scheme achieves traceability.

5.5 Non-repudiation

Suppose the court requires the CA to disclose the suspect Alice's information as evidence. The CA then provides Alice's true identity to the court as evidence. To prevent any party from falsifying evidence or denying that the published results match the certificates they hold, the certificate needs to contain strong evidence that ensures no party can forge or repudiate the results. We divide the

situation into two scenarios: one where the CA is dishonest and one where the user is dishonest.

The first scenario is where the user is dishonest. Alice wants to deny her crime, so she claims that the result published by the CA does not match the certificate she holds to evade conviction. In this case, the CA publishes the certificate containing the user's digital signature $sig(M)$ and identity information $M = (ID_A \,||Alias_A)$. $sig(M)$ is the initial signature generated by Alice for her identity information $M = (ID_A \,||Alias_A)$. This signature has unforgeability, making it impossible for Alice to deny her identity.

The second scenario is where the CA is dishonest and fabricates evidence against Alice. Alice can also use $sig(M)$ to compare with the result published by the CA. Although the CA can verify Alice's digital signature $sig(M)$, it cannot forge Alice's signature. That is, Alice's signature $sig(M)$ is not equal to the CA's forged signature $sig'(M)$. The CA also cannot authenticate Alice's identity to anyone else.

Both of the above scenarios can be defended against, making the results more credible and preventing either party from denying them. Therefore, our scheme can achieve non-repudiation of identity.

5.6 Replay attack

The replay attack is generally used to paralyze system services by sending a large number of service requests that the system can accept without responding, thereby causing the system to become paralyzed. Based on the current user signature and verification process, the attacker must first obtain the content sent by Alice along with her signature before replaying it, which is quite difficult. The most important aspect is that by checking the signature content, if the content is repeated, it can be discarded immediately, and the system does not need to wait for a response. Therefore, concerns about replay attacks can be ignored. Additionally, a nonce can be included to check the response to further enhance system protection.

5.7 Comparison

This paper is based on the Elliptic Curve Discrete Logarithm Problem and bilinear pairing, and it protects users' identities from being identified and stolen. The parameters corresponding to the real information of the users in the system are secret and not publicly disclosed, while the parameters in the protocol are controllable but cannot be forged. On the other hand, when necessary, the court can summon evidence provided by the CA to disclose the true information of the certificate holder. The parameters corresponding to user information cannot be forged, and users cannot deny them, making the protocol in this paper more credible. We will present some security items and use Table 3 to show the comparison between the schemes in previous literature and the scheme in this paper.

Table 3. Functional Comparison Between This Paper and Other Protocols

	Rivest et al Scheme [14]	Rabadi Scheme [12]	Proposed Scheme
A1	✓	✓	✓
A2	✓	✓	✓

A3	✓	✓
A4		✓

Source: By authors.

A1. Identity Protection

A2. Unforgeability of Signature and Identity

A3. Effective Traceability of User's True Identity

A4. Non-repudiation of True Identity

6. Conclusion

We analyzed past techniques for online identity anonymity and related laws, and found that most research focuses on concealing user identities. The drawback is that utilizing anonymous and altered identities in cybercrime can make investigations and law enforcement difficult. Implementing real-name authentication for users can regulate cybercriminal behavior, but it may lead to other issues, such as privacy violations. We hope to protect identity privacy while not compromising public acceptance due to privacy concerns. Therefore, this study proposes an anonymous certificate application protocol with traceability, based on Public Key Infrastructure (PKI) and elliptic curve cryptography techniques to protect identity privacy, while maintaining identity records to enhance traceability.

This study achieves the requirements of anonymity, identity authentication, and non-repudiation of identity. Users register their true identities with the certificate authority (CA), which then issues an anonymous certificate endorsed by the CA. We embed identity records into the certificate, hiding the real data to protect personal identity privacy. Anyone can verify the validity of the certificate, but cannot directly identify the specific individual from the certificate's information. If the holder engages in cybercriminal activities using anonymity, the CA can trace the holder's identity information through the retained records, preventing the holder from evading justice due to insufficient evidence. This not only achieves the anonymity feature of concealing identity but also realizes the traceability of a real-name system, balancing the convenience and security of the internet, and meeting the needs of internet freedom and identity recognition.

References

[1] Yang, W.C. et al., Final Report on the Behavior Patterns and Prevention Technologies of Internet Anonymization and Identity Forgery. National Communications Commission Research Report. Available online: https://www.ncc.gov.tw/chinese/news.aspx?site_content_sn=1812, 2011.11.

[2] National Regulations Database, Enforcement Rules of the Personal Data Protection Act. Available online: https://law.moj.gov.tw/LawClass/LawAll.aspx?pcode=I0050022, 2016.03.

[3] ITU-T X.509, “Information technology - Open Systems Interconnection - The Directory: Public-key and attribute certificate frameworks,” https://www.itu.int/rec/T-REC-X.509-201910-I/en, 2019.

[4] Alagic, G. et al. "Status Report on the Third Round of the NIST Postquantum Cryptography Standardization Process," NIST Special Publication, NISTIR 8413, 2022. DOI: 10.6028/NIST.IR.8413.

[5] Taiwan Government Public Key Infrastructure. Available online: https://grca.nat.gov.tw/index2.html, Query Date: 2024.03.20.

[6] Koblitz, N. Elliptic curve cryptosystems. Mathematics of Computation, 1987, 48 (177), 203-209.

[7] Victor S. Miller. Use of elliptic curves in cryptography. CRYPTO 85, 1985, 417-426.

[8] National Institute of Standards and Technology(NIST). Available online: http://csrc.nist.gov/groups/ST/toolkit/documents/SP800-57Part1_3-8-07.pdf, Query Date: 2020.07.18.

[9] Joux, A. The Weil and Tate Pairings as Building Blocks for Public Key Cryptosystems. in Proceedings Fifth Algorithmic Number Theory Symposium. Lecture Notes in Computer Science, Springer-Verlag, 2002.

[10] Boneh, D., Lynn, B. and Shacham, H. Short Signatures from the Weil Pairing. Advances in Cryptology-Asiacrypt'01, 2001, LNCS 2248, 514-532.

[11] SedatAkleylek, Barıs Bulent Kırlar, Omer Sever and ZalihaY¨uce. Short Signature Scheme From Bilinear Pairings. Journal of telecommunication and information technology, 2011.

[12] Nader M. Rabadi. Anonymous Group Implicit Certificate Scheme. Consumer Communications and Networking Conference (CCNC), 2010.

[13] Brown, D.R.L., Gallant, R.P. and Vanstone, S.A. Provably secure implicit certificate schemes. Financial Cryptography, 2002, LNCS, 2339.

[14] Rivest, R.L., Shamir, A. and Tauman, Y. How to leak a secret. Advances in Cryptology-Asiacrypt 2001, 2001, LNCS 2248, 552-565.

[15] Yang, W.C. and Huang, Z.Z. Traceable Alias Protocol based on Implicit Certificates. The 24th National Information Security Conference, 2014, 05.

[16] Chen, L., Moody, D., Randall, K., Regenscheid, A. and Robinson, A. Recommendations for Discrete Logarithm-based Cryptography: Elliptic Curve Domain Parameters. NIST Special Publication, NIST SP 800-186, 2023. DOI: 10.6028/NIST.SP.800-186.

[17] SECG. SEC 1: Elliptic Curve Cryptography, May 2009. Version 2.0. www.secg.org.

[18] Markel, A. and Nemirovskiy, L. Pairing-based short signatures. https:// markel.co/projects/ecc/2/article.pdf, 2014, accessed 14 January 2024.