Secure Identity Based Encryption Without Random Oracles

Design:

Identity-based encryption is a cryptographic scheme that allows encryption and decryption based on identities, such as email addresses or usernames, rather than traditional public keys. The Boneh-Boyen IBE scheme relies on bilinear pairings over elliptic curve groups and admissible hash functions. The setup phase generates system parameters and a master secret key. Key generation utilizes a random hash function key to hash identities to {0,1}^n. Encryption and decryption operations leverage bilinear pairings for efficiency.

What are Bilinear Pairings?

Bilinear pairings are mathematical operations defined on groups that have specific properties. In the context of cryptographic pairings, we deal with groups that have a bilinear map defined between them.

Groups:

- In the context of pairings, the groups denoted as G_1,G_2 and G_T.
- These groups have different properties but are typically defined over elliptic curves or finite fields.

Bilinear Map:

- A bilinear map is a function that takes elements from two different groups and maps them to an element in a third group.
- Formally, a bilinear map e:$G_1 \times G_2 \rightarrow G_T$ satisfies the following properties:
    - Bilinearity: For all $P,Q \in G_1$, $R,S \in G_2$, and $a,b \in Z$:

$$e(aP,bQ) = e(P,Q)^{ab} = e(P,bQ)^{a} = e(aP,Q)^{b}$$

    - Non-degeneracy: There exist elements $P \in G_1$ and $Q \in G_2$ such that $e(P,Q) \neq 1$

1. **Key Generation**:
    - In IBE schemes, a key generator computes private keys based on the identities of users. Bilinear pairings enable the generation of private keys such that they are uniquely associated with user identities.
2. **Encryption**:
    - Bilinear pairings enable the encryption process by providing a way to compute ciphertexts that are dependent on both the plaintext message and the identity of the recipient.
    - By using bilinear pairings, IBE schemes can ensure that ciphertexts are bound to specific user identities, allowing only the intended recipients to decrypt them.
3. **Decryption**:
    - During decryption, bilinear pairings enable the computation of a decryption key that can effectively recover the original plaintext message.

- By exploiting the properties of bilinear pairings, IBE schemes ensure that only users possessing the correct private key associated with the ciphertext's identity can successfully decrypt the message.

Implementation:

The implementation consists of the following classes and methods:

- Boneh_Boyen_IBE: This class contains methods for setup, key generation, encryption, and decryption.
- setup(): This method generates system parameters and a master secret key.
- KeyGen(params, ID, master_key): This method generates a private key for a given identity.
- encrypt(params, ID, M): This method encrypts a message for a given identity.
- decrypt(params, dID, cipher_text): This method decrypts the ciphertext using the private key.

System Parameters and Master Secret Key Generation (Setup Algorithm):

The setup algorithm generates system parameters and the master secret key.

System parameters include:

- g: a random generator in G_1
- $g_1 = g\alpha$: where α is a random element from Zp
- g_2: a random generator in G_2
- U: an $n\ x\ s$ matrix where each element $u_{i,j}$ is a random element in G_2
- k: a random hash function key chosen from the family of hash functions
- ${g_2}^{\alpha}$: master key

Private Key Generation (KeyGen Algorithm):

- The KeyGen algorithm generates a private key for a given identity ID
- It hashes the identity to $\{0, 1\}^n$ based on the group and length n.
- It chooses a random r from Zp.

Encryption (Encrypt Algorithm):

- The encrypt algorithm encrypts a message M for a given identity ID.
- It hashes the identity to $\{0, 1\}^n$ based on the group and length n.
- It picks a random t from Zp.
- It performs the following operations:
  1. $A = M * e^t$,where e is the pairing between g_1 and g_2
  2. $B = g^t$
  3. $C_i = u_{i,a_i}^t$,for each i

Decryption (Decrypt Algorithm):

- The decrypt algorithm decrypts the ciphertext using the private key.
- It calculates the decrypted message M as follows:

$$A \cdot \frac{\prod_{j=1}^{n} e(C_j, d_j)}{e(B, d_0)}$$

Security Analysis:

Selective identity model: adversaries can adaptively choose the identities for which they wish to obtain secret keys. This means that the adversary can interact with the system, requesting secret keys for any identities of its choice. This means that even if the adversary obtains secret keys for specific identities, the security of the scheme remains intact.

Semantic Security: The Boneh-Boyen IBE scheme provides semantic security, meaning that an adversary cannot distinguish between encryptions of different messages under the same identity. Even if the adversary obtains secret keys for specific identities, they cannot use this information to decrypt messages encrypted under other identities.

Hardness Assumptions: The security of the Boneh-Boyen IBE scheme relies on the hardness assumptions associated with bilinear pairings, particularly the Decisional Bilinear Diffie-Hellman (DBDH) assumption. The security proof demonstrates that breaking the scheme is computationally difficult, even when the adversary has obtained secret keys for certain identities.It says given the elements $g, g^a, g^b, g^c, e(g,g)^{abc}$ for a random $a, b, c \in Z_p^*$, it is computationally hard to distinguish $e(g,g)^{abc}$ from a random element in target group G_T .In simpler terms, the DBDH assumption claims that given certain pairs of elements generated using the bilinear map, it is computationally difficult to determine whether a specific value is the result of a bilinear map computation or just a random value in the target group.