Related Work

We let λ be the security parameter given to the setup algorithm such that a security level of 256 bits is realised. We let G be some BDH parameter generator. S is the set of desired recipients i with $i = 1, .., n$. Symmetric encryption and decryption is done using AES Galois Counter Mode $GCM_{enc}(P, A, K, IV)$ and $GCM_{dec}(P, A, K, IV)$ respectively.

Setup(λ)***:*** Given a security parameter $\lambda \in \mathbb{Z}^+$, the algorithm works as follows:

1. Run G on input λ to generate a prime q, two groups $\mathbb{G}_1, \mathbb{G}_2$ of order q, and an admissible bilinear map $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$. Choose a random generator $P \in \mathbb{G}_1$
2. Pick a random $s \in \mathbb{Z}_q^*$ and set $P_{pub} = sP$
3. Choose a cryptographic hash function $H_1 : \{0,1\}^* \to \mathbb{G}_1^*$ and $H_2 : \mathbb{G}_T^* \to \{0,1\}^{256}$. The security analysis will view H_1, H_2 as random oracles.

The symmetric key space is $K = \{0,1\}^{256} = \{K_1||IV\}$ with $K_1 = \{0,1\}^{128}$ and $IV = \{0,1\}^{128}$. The ciphertext space is $C_i = \mathbb{G}_1^* \times \{0,1\}^{256}$. The system parameters are $params = \{q, \mathbb{G}_1, \mathbb{G}_2, e, P, P_{pub}, H_1, H_2\}$. The master key is $s \in \mathbb{Z}_q^*$.

KeyGen(λ, ID_i)***:*** For a given string $ID_i \in \{0,1\}^*$ the algorithm does:

1. Compute $Q_{\mathrm{ID}_i} = H_1(ID_i)$
2. Set the private key d_{ID_i} to be $d_{\mathrm{ID}_i} = sQ_{\mathrm{ID}_i}$ where s is the master key.
3. Return d_{ID_i} to the corresponding user ID_i over a secure channel.

Encrypt$(params, \lambda, K, S)$***:*** To encrypt K under the public keys $\{ID_i \in S\}$:

1. Generate a random symmetric session key $K_1 = \{0,1\}^{128}$. Generate a random initialisation vector $IV = \{0,1\}^{128}$ and set $K = \{K_1||IV\}$
2. Choose a random $r \in \mathbb{Z}_q^*$
3. For each recipient $ID_i \in S, i = 1..n$, calculate the ciphertext
$$C_i = K \oplus H_2\left(g_{\mathrm{ID}_i}^r\right) \quad \text{where} \quad g_{\mathrm{ID}_i} = e\left(Q_{\mathrm{ID}_i}, P_{pub}\right) \in \mathbb{G}_T^*$$
4. Apply GCM with initialisation vector IV and secret key K_1. Plaintext is set to $P_{text} = K$ and the additional authenticated data $A = \{rP||C_1||C_2||..||C_n\}$. GCM then outputs a ciphertext C_T and an authentication tag T such that
$$\{C_T, T\} = GCM_{enc}(P_{text}, A, K_1, IV)$$
5. The following message is then broadcasted over an insecure network
$$M = \{C_T||A||T\}$$

Decrypt$(params, d_{ID_i}, M)$***:*** Parse broadcasted message M as $\{C_T||A||T\}$. For each $C_i \in A = \{rP||C_1||C_2||..||C_n\}$ do the following:

1. Decrypt C_i using the private key d_{ID_i} by calculating

$$C_i \oplus H_2\left(e\left(d_{\mathrm{ID}}, rP\right)\right) = K = \{K_1||IV\}$$

2. Decrypt C_T by

$$\{P_{text}, T_{dec}\} = GCM_{dec}\left(C_T, A, K_1, IV\right)$$

3. Verify whether T_{dec} corresponds to T in M.
   If $T \neq T_{dec}$, try next C_i.
   When all C_i are parsed $(i = n)$ and still $T \neq T_{dec}$, return $\perp$.