Security and Privacy Issues of IoT at Fog Layer Architecture

Vinay Michael (vinaymachan@gmail.com)
Charles Sturt University CSU: Charles Sturt University https://orcid.org/0000-0003-3415-1264

Research Article

Keywords: Fog Computing, Internet of Things, Cipher-text attribute-based encryption, Cipher-text update, Quantum cryptography, bilinear operation

Posted Date: March 23rd, 2021

DOI: https://doi.org/10.21203/rs.3.rs-242841/v1

Security and Privacy issues of IoT at Fog layer architecture

Author: Vinay Michael, Email: vinaymachan@gmail.com

Abstract:

Internet of Things (IoT) based applications and systems are gaining attention in the recent days because of their vast benefits such as efficient utilization of resources, enhanced data collection, improved security, lesser human efforts and reduced time. Security of sensitive data in IoT based fog environments is inevitable to prevent those data to be misused by the attackers. In this study, we present an improved hybrid algorithm termed as HQCP-ABE (Hybrid Quantum key Cipher text Policy Attribute based Encryption with Cipher text update) that integrates highly effective algorithms such as CP-ABE, Quantum key cryptography and cipher text update. The proposed algorithm eliminates the need of costly pairing during decryptions and efficiently performs data encryption, decryption and user authorization. The proposed protocol is demonstrated to be highly efficient in terms of encryption and decryption while compared to other existing methods. It also achieves lesser packet loss, reduced control overheads, reduced computational overhead during encryption and decryption processes, lesser delay, improved security, packet delivery ratio, throughput, network lifetime with limited bandwidth and user privacy. We further considered energy consumption in this study. The proposed HQCP-ABE method is demonstrated using ns3 simulation and compared with existing CP-ABE and PA-CPABE methods.

Keywords: Fog Computing, Internet of Things, Cipher-text attribute-based encryption, Cipher-text update, Quantum cryptography, bilinear operation

Introduction:

In the recent days, the substantial development in Internet of Things (IoT) technology and the number of users utilizing internet had enabled large amount of users to create, transmit and save massive amount of data and sensitive information. IoT based systems grabbed considerable attention across the globe towards furnishing solutions for several real-time applications in healthcare, smart grid, agriculture, industry, home, transport and many more. Any conventional device or appliance can be revolutionized into smart IoT based device by authorizing inter-communication protocols and internet connectivity [1]. To provide higher level of quality services, data received from the IoT devices is forwarded to a centralized location and treated as per the application requirements. Managing vast amount of data for transfer and processing through cloud computing resulted in various problems namely increased response time and delayed services. Further, meeting out the demand in the real-time data transfer and processing due to the increased number of applications in IoT, the need for IoT is significantly increasing [2]. Fog computing emerged as a computing model for data handling by directing near field communication with sensors. Since Fog computing employs near field communication, the response time of the model is quicker and the number of sensors assigned to a device is lesser while compared to cloud computing [3-5]. Fog based IoT is an innovative and remarkable communication technology solution that offers services with unchanging network bandwidth, portability assistance, reduced latency and location recognition. Fog based communications also offers reliable and trustworthy solutions to escort the fog resources and services adjacent to the users and further supports in utilizing the available resources and services in the network edge [6]. Fog computing considers data transfer and utilization as significant. Similarly, data transfer and access to a legitimate user is most important. Therefore, data security [7] is a demanding requirement to ensure the security of sensitive information specifically in today's advanced communicating technology and innovative trends namely in fog based IoT networks. But, various issues breach data privacy during data transmission and storage. These challenges pose great threat to data security of sensitive information. Similarly, the data transmission of sensitive information through insecure mediums in case of fog based IoT networks becomes an issue [8] that has to be focused instantly. Various encryption techniques were presented earlier in regard to constructing data transmission and storage functional areas which are categorized into partial or complete encryption methods. Encryption methods involve in either completely or partially encrypting data through conventional block cipher methods namely AES, DES or stream cipher methods. Cipher-text Policy attribute based encryption (CP-ABE) is identified as an effective encryption technique for outsourcing data [9]. Though CP-ABE offers higher flexibility and scalability in establishing secure data access control in outsourcing data, it suffers from a major issue called revocation management. Various CP-ABE methods were proposed to resolve the revocation problem. Proxy re-encryption technique has been identified as an effective method [10] to resolve the revocation problem in CP-ABE. Proxy re-encryption (PRE) method envoys the function for re-encryption to a semi-trusted proxy and decreases revocation costs and updates of the policies but creates more overheads for data owners. To resolve the challenges in the PRE methods, more specifically in handling sensitive information in the internet based environments, Quantum information processing method [11] plays a significant role. Quantum Cryptography aims in transmitting quantum information through open channels. This method grabbed a considerable attention among researchers. So, in this paper, we investigate the issues in the conventional approaches by presenting a hybrid approach to improve the data security and user privacy by improving the network lifetime utilizing limited bandwidth. Furthermore, we evaluate the feasibility and convergence of our proposed hybrid approach through simulations and experimental trials.

The main contributions of this research study can be summarized as follows:

(i) To provide higher accuracy in sensitive data protection and data confidentiality, we formulate a hybrid approach termed HQCP-ABE (Hybrid Quantum key Cipher text Policy Attribute based Encryption with Cipher text update) which is an integration of three conventional, well-known approaches as CP-ABE, Quantum key Cryptography and the extension method of CP-ABE, cipher-text update [12]. The proposed hybrid method overcomes all the drawbacks in CP-ABE, cipher-text update process and Quantum cryptography schemes and provides effective encryption and decryption of the valuable data and guarantees data security and confidentiality.

(ii) The formulated hybrid approach HQCP-ABE solves the problem in the cipher-text update process by integrating Quantum cryptography method. It avoids the costs incurred during bilinear pairing operations during decryption process in the cipher text update process.

(iii) The incorporation of cipher-text update and computation outsourcing scheme in the hybrid algorithm satisfies other needs of data owners who could like to authorize some other users in updating the encrypted data. Similarly, the addition of Quantum key Cryptography method in the hybrid algorithm takes over the major computing processes for encryption, decryption and user authorization.

The rest of this study is arranged as follows. The related studies are reviewed in Section 2. Section 3 describes the system overview and model for our proposed system and the key approaches used. Section 4 presents the security driven HQCP-ABE approach. Section 5 discusses about the algorithm constructions for our proposed system. Section 6 discusses about the analysis of security. Simulation results and discussions were analysed in Section 7. Finally, we conclude this study in Section 8.

2. Related Works

The technological advancement of IoT techniques in Fog based systems in the recent days gives rise to various recommendation approaches for Fog based IoT systems. This section discusses about the research studies in several literatures.

2.1. ABE based methods for Secure Fog communications:

Fog computing has been introduced by Cisco in 2014 which is considered as a layer that present in the centre of the cloud where the target users contain fog nodes namely switches, routers etc (Alrawais et.al 2017). They are very near to the target users compared to that of cloud servers and also few of the tasks and services offered in the cloud were transferred to the fog nodes. Alike cloud servers, fog nodes also do not offer complete trust. Data security would give rise to considerable distress from the users while they store valuable information over cloud servers using fog nodes (Lee et.al 2015). Hence, recognizing a novel access control method to manage the cloud, fog and its users is required because they both contain varying system models and network structures. Moreover, the recognized model should decrease the computational complexity and increase larger flexibility for the fog users.

Attribute based Encryption (ABE) is recognized as a symmetric key encryption method that is capable of providing highly scalable, flexible and fine grained access control to the designed environment. This method was modelled and developed by (Sahai et.al 2005 and Lewko et.al 2011). ABE method has been classified into Cipher-text policy ABE (CP-ABE) and Key-policy ABE (KP-ABE) defined in (Sahai et.al 2007 and Goyal et.al 2006). This method has been very successful and robust in realizing access control in various applications of IoT (Ruj et.al 2011, Hu et.al 2016 and Jiang et.al 2017). (Yu.et.al 2011) utilized KP-ABE method for data protection to deal the fine grained data access control issue in WSNs. Because of the eloquence in detailing access control policy of the cipher-text, CP-ABE method is ideal for access control in IoT based systems while compared to KP-ABE. In the real-world applications, ABE methods face various shortcomings. Specifically, they consume more computational costs which give rise to become unrealistic in running the ABE based algorithms for resource constrained devices. Further, ABE methods (Sahai et.al 2007 and Jiang et.al 2018) are constructed through bilinear pairing operations and hence their algorithms for decryption operation needs more expensive pairing where one pairing here takes three times or larger of one exponentiation operation. To resolve this issue, (Green et.al 2011) presented a method that re-distributes the decryption process workload in the ABE method to a server (or a proxy) in which the private key(attribute) is bisected into decryption key for data user and transformation key for the proxy server. In this method, it takes single exponentiation process by the data user for decrypting the result obtained from the proxy server to get the actual message. But, the proxy server may fail to perform proper calculations since it is not trusted. (Lai.et al 2016) presented ABE method with verifiable and outsourced decryption scheme (ABE-VOD) to address the proxy server issue but this scheme contributes more calculations to the actual ABE method. (Alrawais et.al 2017) presented a security approach in fog computing to assure the basic requirements to initiate communication among the cloud and the fog nodes namely authorization, access control, privacy, and conformity. CP-ABE method is used to ensure authorization and privacy among the cloud and the fog nodes. But, this method resulted in larger computational complexity and overheads. To address these issues, (Huang et.al 2017) proposed a secure data access control method for fog aware IoT systems. Initially, the user's sensitive data are encrypted with a novel update policy with the conventional access policy. Further, the data is re-distributed to the cloud servers using fog nodes. In this method, the attributes of the authorized users that persuade the access policy are capable of decrypting the cipher-text. To avoid modification of data by the attackers, this method will check for the signature as well to renew the cipher-text. This method ideally suits for resource limited IoT devices in fog computing but requires expensive bilinear pairings during encryption and decryption.

2.2. PRE based methods for Secure Fog communications:

To resolve the issues in ABE methods, Proxy re-encryption can be used as an effective solution which is used in various cloud applications. Though various PRE methods were proposed (Libert et.al 2008, Fang et.al 2009, Fang et.al 2012), all these methods let the proxy to transform Alice's cipher-text at a time for one delegate. (Weng et.al 2009) presented conditional PRE to ensure flexible control over the encrypted cipher-text where the cipher-texts that fulfils the conditions fixed by Alice are re-encrypted. But, the proxy fails to do re-encryption for a set of users at a time for the cipher-text. To address this problem, (Sun et.al 2016) proposed proxy broadcast re-encryption method where a set of users for the Alice's cipher-text can be encrypted at a time. Still, this model is not a standard one and also requires spending of bilinear pairing costs. (Khashan et.al 2020) constructed a systematic cryptographic interpretation to ensure increased security for establishing communication among fog to things using Proxy re-encryption. To defeat the challenges related to privacy and security, different security methods were utilized for transmitting, storing and retrieving data information in fog computing. But, these methods could end up in larger computational overheads, control the validity of devices, and show consequential latency for time-sensitive applications which requires response. This method is not ideal for real time IoT scenarios.

2.3. Quantum Cryptography based methods for Secure Fog communications:

Quantum cryptography method forwards quantum or classical data through open channels. They are broadly categorized into four types depending on the type of embedding technique being used: quantum data hiding method (QDH)(Vincenzo et.al 2002 and Leung et.al 2005), quantum Cryptography(Atty et.al 2017 and Jiang et.al 2015), quantum Cryptography protocols (Martin 2007 and Liao et.al 2010) and quantum error correcting code (QECC)(Banacloche 2002 and Shaw et.al 2011). (Banacloche 2002) proposed a quantum based protocol which hides secret information as an error. (Shaw et.al 2011) proposed a quantum based protocol which employs noisy quantum channels but it takes an additional Bell state for every four bit secret information that is transmitted. (Matin et.al 2007) proposed a quantum based protocol for cryptographic communications using BB84 protocol (Bennett et.al 2014). (Liao et.al 2010) achieved sharing of secret information using multi-party quantum Cryptography communication. To achieve higher level of security, (Latif et.al 2018) proposed a new quantum Cryptography method for securing messages in the fog aware IoT environment. This method uses quantum entangled states, XOR operation, hash function and gray code for reliable and trustworthy communication.

The above discussed approaches aim to secure sensitive information using various techniques. These approaches will be further efficient if they are integrated as a hybrid approach which ensures higher security and data confidentiality. Hence, in this study, we propose a hybrid approach called HQCP-ABE by integrating the well-known CP-ABE, CP-ABE with cipher-text update and Quantum cryptography with one time pad (OTP) mechanism.

3. System Overview

The system overview of our proposed scheme is shown in Fig. 1. It constitutes controlling attributes, control server(cloud), fog nodes, data owners and users.

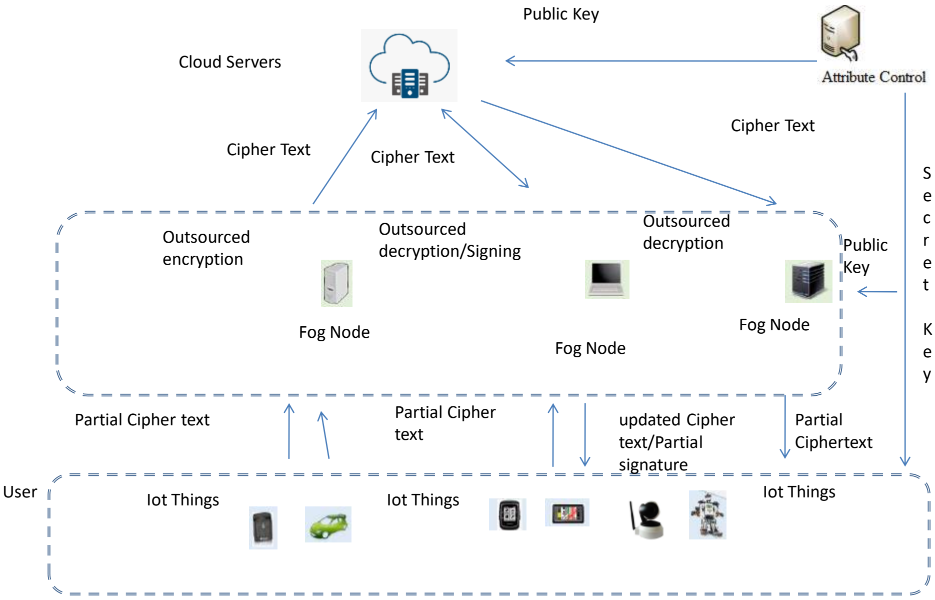


Fig 1: System Model

1) Controlling of Attributes:

Controlling attributes is completely trustworthy and it is responsible for producing system parameters and also the secret key for every user.

2) Control Server:

The Control server provides the storage service through online and it is a semi-trusted device. Signature verification and cipher-text updating process is accomplished by a control server.

3) Fog node:

The fog nodes that are placed over the network edge provide number of services. It is responsible for generating cipher-text and transferring cipher-text to control server. It also decrypts the cipher-text for users. Further, it helps end users for signing the cipher-text update demand.

4) Data owner:

Data owner is responsible for uploading IoT things information to the cloud server. It is used for accessing and updating policies for generating entire cipher-text with fog nodes.

5) User:

Users which are connected to the fog nodes contain IoT devices like smart meters, smart cameras and medical sensors. Due to limited storage and computation ability in IoT devices, fog nodes furnish necessary assistance for accessing cipher-text which is saved in the control server. If the attributes set furnished by the user satisfies the access policy present in the cipher-text, they are allowed for decrypting the data. If a user wishes to perform any alteration and needs to re-encrypt the data after accessing the data, control sever regenerates the cipher-text stored for the user only if their attributes set assures the update policy in the cipher-text.

3.1. Preludes and Descriptions:

i. Bi-linear mapping:

Assume M_a and M_b as two product categories that are prime pr. A bi-linear map can be defined as a function, $k: M_a X M_a \rightarrow M_b$ and contain the characteristics as follows:

1. **Computing ability:** An effective algorithm is presented to calculate $k(i,j) \in M$, for every $i,j \in M$
2. **Bilinear operation:** For every $i,j \in M$ and $\propto, \beta \in Y_{pr}$, it contains $k\left(i^{\propto}, j^{\beta}\right) = k(i,j)^{\propto\beta}$
3. **Generating ability:** If d is assumed as a generator of M, then $k(d,d)$ will also be considered as a generator of M_1.

3.2 System initialization and setup of the certifying authority

The System initialization setup $init_{setup(ini^{\partial})}$., CC requests to select system parameters. The authorized security parameters ∂ as the security variable for input and produces public key $\mathrm{PU}_{\mathrm{key}}$ and the master key $\mathrm{MS}_{\mathrm{key}}$ as output. The authority for certificate starts the system using the $CertAuth$ algorithm as follows:

$$CertAuth(1^{\partial}) \rightarrow (\mathrm{MS}_{\mathrm{key}}\mathrm{PS}\ (sigk_{CertAuth}, verk_{CertAuth})). \tag{1}$$

The above statement considers parameter measures ∂ as inputs and delivers the system master key $\mathrm{MS}_{\mathrm{key}}$, public variables PS along with a signature pair and a verifying key $(sigk_{CertAuth}, verk_{CertAuth})$ as outputs.

(ii) Registering the users:

The users in the fog system forward their identification information to the certifying authority $CertAuth$. Executes the $RegUser$ algorithm:

$RegUser(\mathrm{PS}, sigk_{CertAuth}, inf_u) \rightarrow$ $(userid, G\mathrm{PU}_{\mathrm{key}_{userid}}, \mathrm{GSEC}_{\mathrm{key}_{userid},}\ Cert(userid)$ to calculate and revert distinct identification $userid$ for every user, the global public key $G\mathrm{PU}_{\mathrm{key}_{userid}} = g^{userid}$, a global secret key $\mathrm{GSEC}_{\mathrm{key}_{userid}} = o^{userid}$ along with the certificate for the user, $Cert(userid) = Sign_{sigk_{CertAuth}}(userid, u^{userid}, g^{\frac{1}{z}userid})$.

(iii) Registering authorized attributes:

Every authorized attribute forwards their identification information to the $CertAuth$ to get their unique identification, u_{id}.

(iv) Setting up authorized attributes $AuthAtt$:

Every $AuthAtt_{u_{id}}$, $u_{id} \in Si_A$ initialize using the $AuthAtt$ setup algorithm.

$AuthAttSetup(PS, u_{id}) \rightarrow (\mathrm{SEC}_{\mathrm{key}_{u_{id}}}, \mathrm{PU}_{\mathrm{key}_{u_{id}}}, \{VERK_{xu_{id}}, \mathrm{PU}_{\mathrm{key}_{xu_{id}}}\})$.

The received outputs namely SEC$_{\text{key}}$= (α_k ,β_k ,δ_k), PU$_{\text{key}_k}$ = (k$(m, m)^{\alpha k}$, g^{1/β_k}, g^{δ_k/β_k}) are the authority keys namely secret and public keys that belong to $AuthAtt_k$ and {$VERK_{xu_{id}}$=ver_{x_k}, PU$_{\text{key}_{xu_{id}}}$ = $g^{verx_k} J(x_k)^{\delta_k}$)} are the public and secret version keys for every attribute x_k monitored by $AuthAtt_k$.

3.3. System Definition

The HQCP-ABE model contains various stages and algorithms as described below:

Stage 1: System initialization and setup

i) $init\_setup(ini^n)$.The authorized attribute considers n as the security variable for input and produces public key PU$_{\text{key}}$ and the master secret key MS$_{\text{key}}$ as output.

Stage 2: Generation of Keys

ii) GenOfKey(PU$_{\text{key}}$, MS$_{\text{key}}$, E). The authorized attribute considers n as the PUBK, MS$_{\text{key}}$, attribute set E as inputs and produces SEC$_{\text{key}}$, secret key for the user as the output. Further, it will send the re-distribution key, SEC$_{\text{key}}'$ to the fog nodes.

Stage 3: Session Key generation using Quantum Cryptography

The QKM in one-time pad (OTP) mechanism, where the session key's feature has identified the data security. The kernel of the conventional Quantum Random Number Generator (QRNG) is responsible for the quality of the produced random number. Moreover, it is prone to contain the fault of correlating for a more extended period and could fail in the localized randomness testing. Employed a quantum random number generator based on arriving time for generating a random number to avoid such defects. Session key generation was generated using QRNG. Algorithm 1 shows generate keys for encryption and decryption to users and CC, respectively. Moreover, it encrypts the data consumption and generates an on each encrypted value using Quantum key management.

Algorithm 1: Generation of Keys

```
Input:   The authorized attribute considers $n$ as the $\mathrm{PU_{key}}$ , attribute set $E$
Output:  produces $\mathrm{SEC_{key}}$, secret key
step 1:  AC chooses s two random large prime numbers $M_a$ and $M_b$
step 2:  Compute $SS_{key}$ by using QRNG
             where $SS_{key} = [SS_{key_1}, SS_{key_2}, \dots SS_{key_s}]$
step 3:  Generate (GenOfKey($\mathrm{PU_{key}}$, $\mathrm{MS_{key}}$, $E$).
step 4:  End
```

GenOfKey($\mathrm{PU_{key}}$, $\mathrm{MS_{key}}$,E): The authorized attribute considers n as the $\mathrm{PU_{key}}$, $\mathrm{MS_{key}}$, attribute set E as inputs and produces $\mathrm{SEC_{key}}$, a secret key for the user as the output. Further, it will send the re-distribution key, $\mathrm{SEC_{key}}'$ to the users.

Stage 4: Design of Quantum Key encryption algorithm:

- The Quantum key-based cryptographic algorithm utilizes a one-time pad (OTP) mechanism that needs the same length of plaintext, Ciphertext, and the key. In this study, a stream cipher-based algorithm is presented. The plaintext size is broken up into various length sequences, which is the same as the keys' length. Considering the first bit of the sequence $SS_{key}||N||r_{sk}||r_n$ .it generates a permutation matrix and session key SS_{key_0}. To ensure enhanced security to authorize identities once establishing the data communication, the key generating model, GenOfKey($\mathrm{PU_{key}}$, $\mathrm{MS_{key}}$, E), utilizes a session key SS_{key} to update secret $L = [l_1, l_2, \dots, l_n]$.which is key generated by QRNG is $SS_{key} = [SS_{key_1}, SS_{key_2}, \dots SS_{key_n}]$

$$L = SS_{key} \oplus L \tag{2}$$

Hence, every L is utilized to protect against impersonation attacks. The addition of the check code can decrease the communication rejection created by the channel noise. The regular sequence can be broken up into four portions, namely SS_{key},N,r_{sk},r_n. Employ L to retrieve $SS_{key_0}{}'$ from N.where N is the authentication sequence generated using XOR of SS_{key} and L, which is

$$N_1 \ = SS_{key_0} \oplus L_1 \oplus SS_{key_1} \, ife = 1;$$

$$N_e = SS_{key_{e-1}} \oplus L_e \oplus SS_{key_e} \, ife \geq 1$$

$$\begin{cases} SS_{key_0} \ = N_1 \oplus SS_{key_0} \oplus L_1 \, ife = 1; \\ SS_{key_e} = N_i \oplus N_{e-1} \oplus L_e \, ife \geq 1 \end{cases} \tag{3}$$

where we consider that if$L_1 = 0, thenSS_{key_0} = 1, else \ SS_{key_0} = 0.$

If a difference is found among $SS_{key}{}'$ and SS_{key} Evaluate the check code option, and if one error is noted in the stated sequence, the data communication is allowed. Hence, the data communication is authorized if it satisfies the following conditions:

a. If r_{sk} complements with key sequence, r_n will not match but compared to SS_{key}, $SS_{key}{}'$ will contain two varying bits alone where there will not be any fault identified, the key will be placed in N.

b. If r_{sk} complements with key sequence, r_{sesk} will not match but compared to SS_{key}. $SS_{key}{}'$ will contain one varying bit alone where there will be fault identified, the key will be placed no faults in N.

Algorithm 2 performing bilinear pairing encryption uses two permutations and diffusion to make encryption results complicated and decrease the computational costs.

a) Quantum Random Number Generator (QRNG) is responsible for generating the session key SS_{key} = [SS_{key_1}, SS_{key_2}, ... SS_{key_s}];

b) The plaintext T contains u bit that is broken up into v data bit fragments. If the length of the final sector is lesser than v, the same will be filled up with zero.

c) Build a permutation matrix l as per the ascending order of L. If the starting bit of the variable L is zero, the last permutation matrix $FPERM$ is l and SS_{key_0} is one, else the permutation matrix $FPERM$ is l^{-1} and SS_{key_0} is Zero;

d) Perform permutation operation T as per the value of $FPERM$ to get the value of t.

e) Perform encryption operation: $ifr = 1, thenU_1 = SS_{key_0} \otimes t_1 \otimes SS_{key_1}$. $ifr > 1, thenU_r$= $U_{r-1} \otimes t_r \otimes SS_{key}$;

f) Now, perform permutation operation Uas per the value of $FPERM$ to get the value of D.

Algorithm 2: Data encryption using Quantum key cryptosystem

```
Input:      plaintext, Initialize the parameters used as follows:
            $SS_{key}$ = [$SS_{key_1}$, $SS_{key_2}$,....$SS_{key_n}$],
             $L = [l_1, l,$ ...... $l_n]$,
               $T = t[u], s = 1, flag = 0$;
Output:     ciphertext

step   1:   $while(u - v)$ <0 do
            $for$(r=0; r<v; v++) do
             $if(r + flag < u), thenT_s$[r] = $t[r + flag * v]$;
            $else$  $T_s$[r] = 0;
            flag = flag+1; s=s+1;

step   2:   Build a permutation matrix $l$ as per the ascending order of $L$.
step   3:   $ifL$[0] =0, then $FPERM = l, SS_{key_0}$ =1;
             $else$ $FPERM = l^{-1}, SS_{key_0}$ =0;
            For every $T_s$= $\{t_{1,}t_{2,}....t_r\}$ do
step    4:  Replace the variable $T_s$ as per the value in $FPERM$
step    5:  $ifs = 1, thenU_1 = SS_{key_0} \oplus t_1 \oplus SS_{key_1}$
               $elseU_r$= $U_{r-1} \oplus t_r \oplus SS_{key_r}$
step    6:  Replace the value of $U$ as per the value in $FPERM$ to get $D$.
step    7:  End
```

The following steps explain the decryption process:

a) Perform reverse permutation D as per the value of $FPERM$ to get the value of U.
b) Perform decryption operation: $ifr = 1, thent_1 = U_1 \otimes \mathrm{SEC}_{\mathrm{key}_0} \otimes \mathrm{SEC}_{\mathrm{key}_{1.}} ifr > 1, t_r = U_{r-1} \otimes U_r \otimes \mathrm{SEC}_{\mathrm{key}_r}$
c) Perform reverse permutation operation t as per the value of $FPERM$ to get the value of T.
d) Cut off the additional zero in the variable T to retrieve the required plaintext result.

Stage 5: Cipher-text update operation:

iii) Fog Signature ($\mathrm{PU}_{\mathrm{key}}, UCT_{up}$, $PC_{up}, ss_{key}{}'$) The fog node considers public key $\mathrm{PU}_{\mathrm{key}}$, the cipher-text update request of the user UCT_{up}, session key of the user $ss_{key}{}'$ as the inputs and delivers the partial signature $\mathrm{PART}_{\mathrm{sig}}{}'$ along with the global key GL_{key} as outputs.

iv) User Signature ($\mathrm{PU}_{\mathrm{key}}, \mathrm{PART}_{\mathrm{sig}}{}', GL_{\mathrm{key}}$)

The user considers $\mathrm{PU}_{\mathrm{key}}$, partial signature $\mathrm{PART}_{\mathrm{sig}}{}'$ and a global key GL_{key} as inputs and delivers the signature $\mathrm{PART}_{\mathrm{sig}}$ as a result.

v) Key Stability Check ($\mathrm{PU}_{\mathrm{key}}, \mathrm{PART}_{\mathrm{sig}}, GL_{\mathrm{key}}$)
The user considers the public key $\mathrm{PU}_{\mathrm{key}}$, global key GL_{key} and signature $\mathrm{PART}_{\mathrm{sig}}$ as input parameters if the GL_{key} satisfies the key stability check conditions. If the condition gets satisfied, it generates output as 1 else it outputs a 0.

vi) Track ($\mathrm{PU}_{\mathrm{key}}, \mathrm{PART}_{\mathrm{sig}}, GL_{\mathrm{key}}$)

If the key stability check conditions get satisfied, it considers the public key $\mathrm{PU}_{\mathrm{key}}$, global key GL_{key} and signature $\mathrm{PART}_{\mathrm{sig}}$ as input parameters and produces user identity $user_{\mathrm{id}}$, otherwise it produces a defeat symbol, ȡ.

3.4. Traceability Model:

To prove the traceability of the proposed mechanism, a game theory has been considered that gets executed between a challenger and an opponent.

a) **Initialization algorithm:** After the initialization process, the challenger forwards the parameters of the public key $\mathrm{PU}_{\mathrm{key}}$ to the opponent.

b) **Key Request:** The opponent requests the challenger to acquire the session keys of the user, $(user_{\mathrm{id1}}, GL_{\mathrm{key1}})$,), ……. $(user_{\mathrm{idn}}, GL_{\mathrm{keyn}})$,).

c) **Forged Key:** The opponent generates $GL_{\mathrm{key}}*$ of the user after the request. By executing the Track$(\mathrm{PU}_{\mathrm{key}}, \mathrm{PART}_{\mathrm{sig}}GL_{\mathrm{key}}*) \rightarrow user_{\mathrm{id}}$ or ȡ and Track$(\mathrm{PU}_{\mathrm{key}}, \mathrm{PART}_{\mathrm{sig}}GL_{\mathrm{key}}*) \in (user_{\mathrm{id1}}, user_{\mathrm{idn}})$, the opponent can succeed in the game which can be expressed as Succ[Track$(\mathrm{PU}_{\mathrm{key}}, \mathrm{PART}_{\mathrm{sig}}GL_{\mathrm{key}}*)$ $\in$ ȡ U $(user_{\mathrm{id1}}, user_{\mathrm{idn}})$. The proposed mechanism will contain the traceability if each of the attackers at polynomial time have a benefit in succeeding the game theory of traceability.

3.5. Key Stability Checking Model:

Key Stability Check $(\mathrm{PU}_{\mathrm{key}}, \mathrm{PART}_{\mathrm{sig}}, GL_{\mathrm{key}}) \rightarrow 0$ or 1. The user GL_{key} will be verified by the trust center TC to verify whether it fulfills the requirements of the key stability checking model and follows some procedures:

a) Verify the key structure $structGL_{\mathrm{key}}$ with $GL_{\mathrm{key}} = \{Key_1', Key_2', M_1, M_{2,}\{Key_{i,1}, Key_{i,2}\}_{key \in GL_{\mathrm{key}}}\}$ and $Key_1' \in I_p^*, Key_2', M_1, M_{2,}\, Key_{i,1}, Key_{i,2} \in$ P.

b) Verify $t(M_{2,}, p) = t(M_{1,},\ p^a)$.

c) Verify $t(Key_2', p^a p^{Key_1'}) = t(p,\ p)^a\, t(M_{2,}M_1{}^{Key_1'}, h))$

d) Verify $\tau_i \in GL_{\mathrm{key}}, g, r, t\ (Key_{i,2}, p) = t(D_{i,}, Key_{i,1}) \neq 1$.

The $structGL_{\text{key}}$ conditions of the key stability checking model outputs 1 if it is satisfied, else it outputs zero.

3.6. Algorithm for Tracking:

Track$(\text{PU}_{\text{key}}, \text{PART}_{\text{sig}} GL_{\text{key}} *) \rightarrow$ $user_{\text{id}}$ or ȡ. The trust center TC executes the algorithm. If the user key $structGL_{\text{key}}$ fails to satisfy the conditions of key stability checking Key Stability Check$(\text{PU}_{\text{key}}, \text{PART}_{\text{sig}}, GL_{\text{key}}) \rightarrow 0$, it produces an output ȡ, else it collects the identity data $user_{\text{id}}$ from $structGL_{\text{key}}$ of the user through the decrypt operation.

3.7. Security Model:

In this proposal, the fog nodes and cloud servers are assumed to be truthful but strange. They together implement the tasks and could conspire to receive the unaccredited data. The proposed security model follows certain aspects as such as:

a) **Data privacy:** The illegitimate users who are not authorized as the deliberate receivers interpreted by the data owner have to be stopped to access the legitimate data.

b) **User authentication:** The user who fails to satisfy the specified cipher-text update policy should also be stopped from updating the cipher-texts.

c) **Fine grained access control:** The data owner should practice meaningful and supple policies so as to allow the users who hold attributes that satisfies the said policies to access and update data alone.

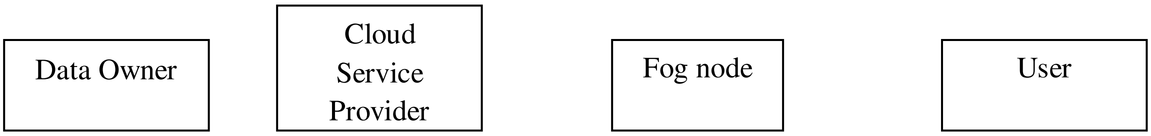
d) **Resistance to Collusion:** Multiple numbers of users are not allowed to merge their outsourcing keys and their secret information to acquire access to the data which cannot be accessed individually.

4. Security driven HQ-CPE method for enhanced data security and user privacy:

In this study, to utilize the benefits of achieving higher levels of security and user privacy, we combine the Cipher-text key policy attribute-based encryption, cipher-text update, quantum key cryptography with one time pad (OTP) mechanism and present an efficient hybrid algorithm HQCP-ABE, which overcomes all the shortfalls of the earlier traditional approaches. In this hybrid algorithm, the computational cost to perform data encryption and decryption sticks to the policy complexity unlike other traditional ABE methods. Moreover, the presented technique avoids costly pairing during decryptions.

The incorporation of cipher-text update and computation outsourcing scheme in the hybrid algorithm satisfies other needs of data owners who could like to authorize some other users in updating the encrypted data. Similarly, the addition of Quantum key Cryptography method in the hybrid algorithm takes over the major computing processes for encryption, decryption and user authorization. Quantum key cryptography method also establishes the overall system decryption precisely. We also measured the energy consumed by the proposed method during encryption and decryption processes to present an energy efficient model.

The entire work flow of the algorithm has been depicted in the work flow diagram below:



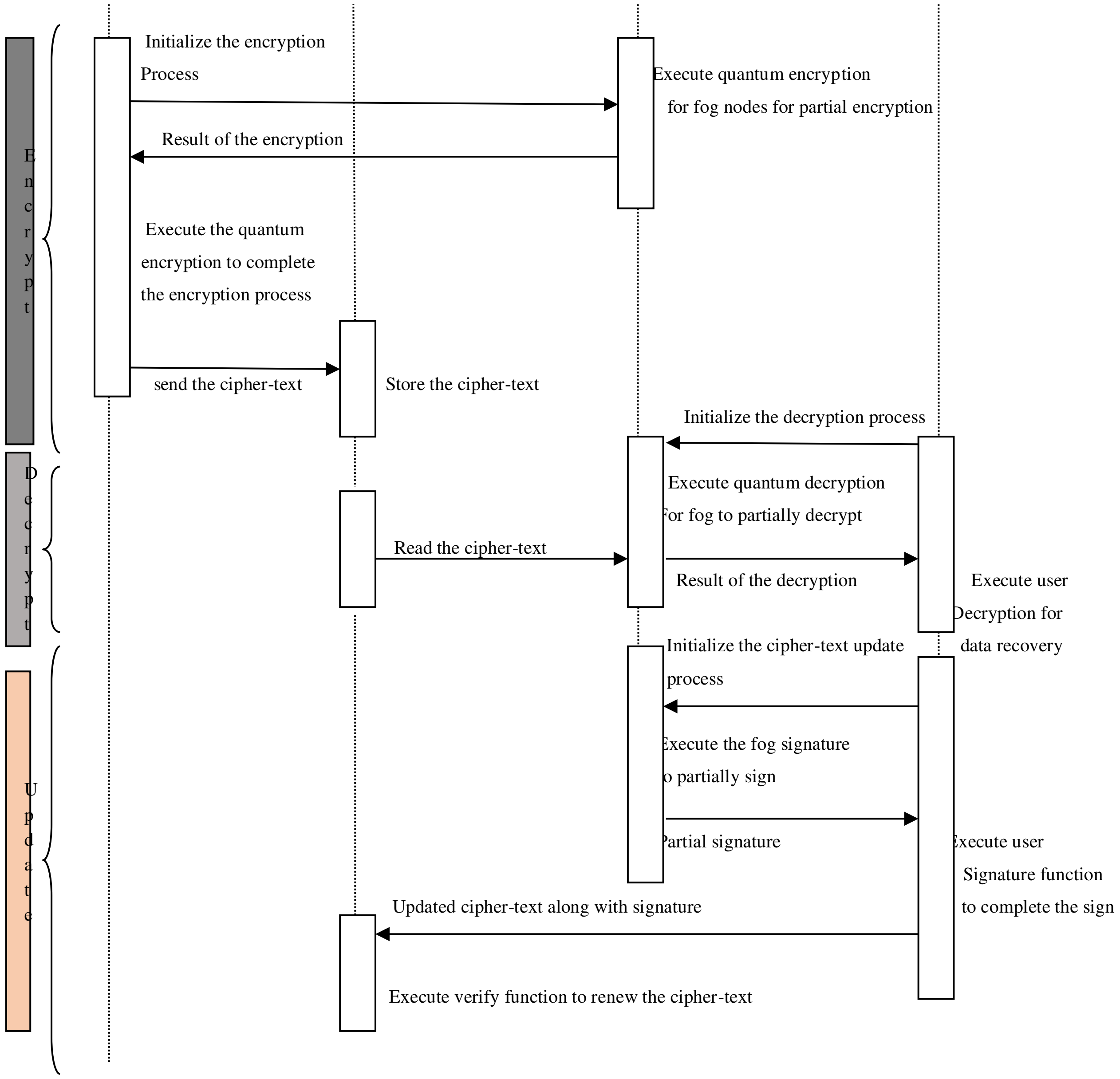

Encrypt
Decrypt
Update
Initialize the encryption Process
Execute quantum encryption for fog nodes for partial encryption
Result of the encryption
Execute the quantum encryption to complete the encryption process
send the cipher-text
Store the cipher-text
Initialize the decryption process
Execute quantum decryption for fog to partially decrypt
Read the cipher-text
Result of the decryption
Execute user Decryption for data recovery
Initialize the cipher-text update process
Execute the fog signature to partially sign
Partial signature
Execute user Signature function to complete the sign
Updated cipher-text along with signature
Execute verify function to renew the cipher-text

Fig.1: Entire Workflow of HQCP-ABE method

The hybrid algorithm takes various steps to establish a highly secure data communication. They are:

i) Pre-processing and set-up

ii) Key generation process

iii) Cipher-text policy update

iv) Master key generation

v) Encryption process

vi) Decryption process

These six steps make the proposed hybrid algorithm highly secure. In the first phase called pre-processing and set-up, a random algorithm (quantum cryptography) will be supplicated by a trusted arbitrator, who is a central authority. It takes security variable as the input, pair of secret keys and public key as the output of attribute dominions. It produces a master secret key and a system public key for the central authority. In the second phase, it will produce user's secret key and a session key by accepting authorized user's secret key as an authorized value. The session key ensures higher level of encryption. The third phase applies cipher-text update process which eliminates the cost of decryption because of bilinear paring in CP-ABE. The master key has been generated by a third party who is trusted. The master key along with the authorized ID of the user in the authority domain will produce the secret key for the user. The fifth phase employs quantum key method for key distribution where two conclaves create a complementary asymmetrical and secret key which is exclusively known to them alone. The sender establishes encryption operation by accepting attribute sets as input for every authority, information or a message to be transmitted and the system public key and outputs the cipher-text. The receiver side performs the decryption process by considering the input (cipher-text). It uses decryption keys for the attribute sets and produces the information or a message as the output to be available for all the authorized users in the receiver side. Hence, we could be able to achieve better security for data and user privacy using the proposed HQCP-ABE approach.

5. ALGORITHM CONSTRUCTIONS:

Due to the resource limitations of IoT devices, fog computing primarily focuses to reduce the computational complexity and overheads of the system. Initially, we present fine grained access control with effective cipher-text update method using quantum cryptography with OTP mechanism and CP-ABE. In this scheme, the attributes of the legitimate users that satisfy the access policy can only be able to decrypt the cipher-text. Similarly, the attributes of those users have to satisfy the update policies can only be able to upgrade the cipher-text. Further, we establish a security aware, energy efficient re-distributable construction that re-distributes many encryption, decryption and signature calculations from the target IoT devices to the fog nodes. The algorithm construction is explained below:

(i) System Setup algorithm:

The authorized attribute executes system setup algorithm to choose the assumed bilinear mapping $k\colon M_a \; X \; M_a \rightarrow M_b$ where M_a and M_b are considered as the two product groups that contain prime order pr and M can be stated as the generator of M_a. Further, the authorized attribute arbitrarily selects $j \in M_a$ and $\propto, \beta \in Y_{pr}$, selects quantum cryptography-based hash functions $J_1\colon \{0,1\} \rightarrow Y_{pr}^{*} \; and \; J_2\colon \{0,1\} \rightarrow M_a$. These functions generates system public key as output, $\mathrm{PUBK} = (\mathrm{m}, \mathrm{j}, \mathrm{m}^{\propto}, \mathrm{m}^{\beta}, \mathrm{j}^{\beta}, k(\mathrm{m}, \mathrm{m})^{\propto\beta})$ as well as the master secret key, $\mathrm{PUBK} = (\propto, \beta)$.

(ii) Generation of Keys

Secondly, the authorized attribute $AuthAtt$ executes GenOfKey algorithm to choose arbitrary value $\delta \in Y_{pr}$, a distinct secret key allocated to every user. Further, every authorized attribute $AuthAtt$ arbitrarily selects $\lambda \in Y_{pr}$ and another arbitrary value, ran_c for every attribute, $C \in E$, where E indicates the user's attribute set. This assumption generates a re-distributable key along with a secret key.

$$\mathrm{SEC_{key}} = (\mathrm{F} = \mathrm{m}^{(\propto+\delta)\beta}) \qquad (1)$$

$$\mathrm{SEC_{key}}' = (F_1 = \mathrm{m}^{\propto}\mathrm{j}^{\lambda}, F_2 = \mathrm{m}^{\lambda}, \{\breve{F_c} = \mathrm{m}^{\delta\beta} \; J_1(\mathrm{c})^{ran_c}), \breve{F_c}' = \mathrm{m}^{ran_c} \} \; C \in E)$$

The re-distributable key, $\mathrm{SEC_{key}}' = (F_1, F_2, \{\widetilde{F_c}, \breve{F_c}'\} \; C \in E)$ of every user will be forwarded to the fog nodes. The authorized user saves the $\mathrm{SEC_{key}}$.

(iii) **Data encryption using Quantum cryptography:**

The data owner initially selects a random key, $ss_{key} \in Y_{pr}$ prior to data upload process to the cloud service provider for the data dat along with ss_{key} using quantum encryption which can be written as $R = SE_{ss_{key}}$(dat). Further, the data owner introduces the access policy AP_t along with the update policy PC_{up} and forwards AP_t to the fog nodes. Then, the fog nodes execute the quantum encryption algorithm to establish the re-distributable algorithm. For every node y in the AP_t, the fog nodes select a polynomial pol_y. The polynomial pol_y is selected in a top-down fashion starting from the root node RT. For every node y in the access policy tree AP_t, fix a degree deg_y for the polynomial pol_y as lesser than the fixed threshold th_y of that particular node. Hence, $deg_y = th_y$ - 1. The quantum key algorithm selects randomly $E \epsilon Y_{pr}$ and fixes $pol_y(0)$ = $pol_{parent(y)}$(index(y)) and selects deg_ypoints arbitrarily to entirely define pol_y. Assume C as the leaf nodes set in the access policy tree AP_t, the fog nodes produce partial cipher-text CPT' as the output. Thus, CPT' = (AP_t, R_3' = $\mathrm{m}^{\beta E}$, R_4' = $\mathrm{f}^{\beta E}$, R_5 ={ $\breve{F}_c{}'$= $\mathrm{m}^{pol_{c(0)}}$, $\breve{F}_c{}'$= $J_1(AuthAtt_c{}^{pol_{c(0)}})$ $_{c\epsilon C}$) (2)

Eventually, the fog nodes remit CPT' back to the data owner. The data owner further executes the quantum encryption algorithm for the data owner explained above to choose $t \epsilon Y_{pr}$ arbitrarily and calculates

R_1= ss_{key}. $k(\mathrm{m},\mathrm{m})^{\propto\beta t}$ along with the Quantum cryptographic session key ss_{key} and calculates R_2= m^t, R_3= R'_3. $\mathrm{m}^{\beta t}$, R'_4. $\mathrm{j}^{\beta t}$. Eventually, the data owner produces the cipher-text CPT as the output.

CPT = (AP_t, PC_{up}, R = $SE_{ss_{key}}$(dat), R_1 = ss_{key}.$k(\mathrm{m},\mathrm{m})^{\alpha\beta t}$,

R_2= m^t, R_3= $\mathrm{m}^{\beta(E+t)}$, R_4 = $\mathrm{f}^{\beta(E+t)}$,

R_5 = {$\widetilde{F_c}$ = $\mathrm{m}^{pol_{c(0)}}$, $\breve{F}_c{}'$= $J_1(AuthAtt_c)^{pol_{c(0)}}$} $_{c\epsilon C}$) (3)

(iv) **Data decryption using Quantum cryptography:**

If the attributes E of the user satisfies the access policy tree AP_t, such user can start the decryption of the cipher-text CPT by executing the algorithm discussed below and get the Quantum cryptographic session key ss_{key}. Then, the fog nodes start to execute the quantum decryption algorithm discussed above to get the cipher-text from the cloud service provider. The decryption algorithm has been written for both the nodes with and without leaf where the former is a recursive algorithm which has to be executed initially. The quantum decryption algorithm considers the cipher-text CPT, $\mathrm{SEC_{key}}'$ and a node y from the access policy tree AP_t as an input.

Scenario 1: Consider $z = AuthAtt_y$ if the node y is a leaf node. If $z \epsilon E$,

$$QuantumDecNode(CPT, \mathrm{SEC_{key}}, y) = \frac{k(D_{\overline{\overline{C}}}, \overline{\overline{F_y}})}{k(D_{\overline{\overline{C'}}}, \overline{\overline{F_{y'}}})}$$

$$= \frac{\boldsymbol{k}(\mathrm{m}^{\delta\beta\, J_1(z)^{rz}}, \mathrm{m}^{pol_{y(0)}}))}{\boldsymbol{k}(\mathrm{m}^{rz}, J_1(AuthAtt_y)^{pol_{y(0)}})} \qquad (4)$$

$$= k(\mathrm{m},\mathrm{m})^{\delta\beta pol_{y(0)}}$$

If $z \notin E$, then $QuantumDecNode(CPT, \mathrm{SEC_{key}}, y) = \perp$

Scenario 2: The quantum decryption algorithm $QuantumDecNode(CPT, \mathrm{SEC_{key}}, y)$ performs decryption as follows if the node y is a non leafy node. Every node y are the child nodes of z, it executes $QuantumDecNode(CPT, \mathrm{SEC_{key}}, n)$ and saves the output in V_n. Consider E_y as a random size of size h_z with children nodes n, $V_n \neq \perp$. If there is no such defined random sized set is present, the node does not get satisfied and hence the defined function returns $\perp$. Else, it calculates and produces the following result.

$$V_z = \prod_{n \in E_y} V_n^{\Delta_{i.E_y(0)}}$$

$$= \prod_{n \in E_y} k(\mathrm{m},\mathrm{m})^{\delta\beta.parent(n)(index(n))})^{\Delta \mathrm{i.Ey(0)}}$$

$$= \prod_{n \in E_y} k(\mathrm{m},\mathrm{m})^{\delta\beta.pol_{y(i)}.\ \Delta \mathrm{i.Ey(0)}}$$

$= \ k(\mathrm{m},\mathrm{m})^{\delta\beta.pol_{y(0)}}$ (5)

Consider i = $index(n)$ and $E_y{}'$ = { $index(n)$: $n \epsilon E_y$. If access policy tree AP_t gets satisfied by E, the outcome of the overall computation for the access policy tree AP_t can be considered as G.

$G = \ QuantumDecNode(CPT, \mathrm{SEC}_{\mathrm{key}}, R)$ = $k(\mathrm{m},\mathrm{m})^{\delta\beta.p_{R(0)}}$

$= \ k(\mathrm{m},\mathrm{m})^{\delta\beta E}$ (6)

Further, the fog nodes calculate,

$A = \frac{k(D_{1,}F_3\ )}{k(D_{2,}F_4\ )} = \frac{k(\mathrm{m}^{\delta}\ \mathrm{j}^{\epsilon}, \mathrm{m}^{\beta(e+t)}\ )}{k(\mathrm{m}^{\epsilon}, \mathrm{j}^{\beta(e+t)}\ )} = \quad k(\mathrm{m},\mathrm{m})^{\delta\beta(e+t)}$ (7) and

$B = A|G = \ k(\mathrm{m},\mathrm{m})^{\delta\beta(e+t)} / \, k(\mathrm{m},\mathrm{m})^{\delta\beta e} \ = k(\mathrm{m},\mathrm{m})^{\delta\beta t}$ (8)

Eventually, the fog nodes forward a partial cipher-text, CP = (AP_t, PC_{up} , R = $SE_{ss_{key}}$(dat) = $ss_{key}.k(SE_{ss_{key}}$, R_2 = m^t , B = $k(\mathrm{m},\mathrm{m})^{\delta\beta t}$ to the fog users. Once getting the value of T from the fog nodes, the fog user executes the quantum decryption algorithm to get the quantum crypt key ss_{key}.

$SS_{key} = \frac{R_{1,}B}{k(R_{2,}F)} = \frac{ss_{key}.k(m,\mathrm{m})^{\alpha\beta t}.\ k(m,\mathrm{m})^{\delta\beta t}\ )}{k(\mathrm{m}^{t}, \mathrm{m}^{\beta(\alpha+\delta)\beta}\ )}$ (9)

Hence, $SE_{ss_{key}}$(dat) has been decrypted using ss_{key} by applying the quantum decryption algorithm.

(v) Cipher-text update process:

Once altering the value of decrypted data, the fog user further encrypts the altered data as explained in the stage of quantum data encryption and further applies signature to the cipher-text update requests using the user's attributes. If the attributes of the fog user present in the signature fulfils the cipher-text update policy, PC_{up}, the cipher-text has been authorized to upgraded by the cloud service provider.

The user forwards a request REQ along with the cipher-text update policy, PC_{up} to the fog nodes. They execute the fog signature algorithm to establish re-distributable signing feature. For every node y present in the cipher-text update policy, PC_{up}, the fog nodes select a polynomial $upol_y$. The polynomial $upol_y$ has been selected from the root node RT in a top-down fashion. Every node y in the tree has been set to the degree deg_y' of the polynomial value $upol_y$ as one lesser than the value of threshold th_y' for that particular node, hence, $deg_y' = th_y'$ - 1. The algorithm selects an arbitrary value r $\in$ Y_{pr} and fix up $upol_y(0) = r$. Further, it selects deg_R' other points of the polynomial value $upol_y$ arbitrarily for defining them. It sets up $upol_y(0) = upol_{parent(y)}(index(y))$ and selects the deg_y' other points arbitrarily to entirely define $upol_y$. Assume Z as the leaf node set in the cipher-text update policy, PC_{up}, the fog nodes deliver the global key GL_{key} as the output.

$$GL_{\text{key}} = \{\widetilde{th_z} = \mathrm{m}^{upol_{z(0)}}, \widetilde{th_{cz}}' = J_1(AuthAtt_z)^{upol_{z(0)}}\}_{z\epsilon Z})) \qquad (10)$$

For every attribute, $h \in Z$, the fog nodes arbitrarily select $t_h \in Y_{pr}$ and calculate using ss_{key}'.

(a) If $h \in E \frown Z$, it calculates as:

$$\widetilde{E_h} = (\widetilde{F_h}.\, J_1(h)^{t_h})^{1/r} = \mathrm{m}^{\delta\beta/r} J_1(h)^{t_h + r_{h/r}},$$

$$\widetilde{E_h}' = (\widetilde{F_h}'.\mathrm{m}^{th})^{1/r} = \mathrm{m}^{t_h + r_{h/r}} \qquad (11)$$

(b) If $h \in Z/E \frown Z$, it calculates as:

$$\widetilde{E_h} = (J_1(h)^{t_h})^{1/r} = J_1(h)^{t_{h/r}}), \widetilde{E_h}' = (\mathrm{m}^{th})^{1/r}$$

$$= \mathrm{m}^{t_h/r} \qquad (12)$$

Further, the fog nodes arbitrarily choose ƛ $\in Y_{pr}$ and produces $SigV'$ the partial signature as the output.

$$SigV' = (\mathrm{REQ}, E_1' = J_2(\mathrm{REQ})^{\text{ƛ}}, E_2' = \mathrm{m}^{\text{ƛ}},$$

$$E_3 = \{\widetilde{E_h}, \widetilde{E_h}'\}_{h\epsilon Z}) \qquad (13)$$

The fog nodes produce back $SigV'$ to the user. Further, the user executes the user signature algorithm to arbitrarily choose $\mu \in Y_{pr}$ and calculate $E_1 = E_1' . J_2(\mathrm{REQ})^{\lambda}.\mathrm{F}$, $E_2 = E_2' . \mathrm{m}^{\mu}$. Eventually, the user produces the signature $SigV$ as the output.

$SigV = (\ \mathrm{REQ}, E_1 = J_2(\mathrm{REQ})^{\lambda+\mu} . \mathrm{m}^{(\alpha+\delta)\beta}, E_2 = \mathrm{m}^{(\lambda+\mu)}, E_3$ (14)

If the user attributes fulfils the cipher-text update policy PC_{up} which is saved in the first cipher-text, the cloud service provider verifies the signature of the user by executing the verification algorithm. The cloud service provider considers to execute the VerificationNode algorithm which is a recursive algorithm. This algorithm considers the signature $SigV$, the global keyGL_{key}, node y from the updating tree PC_{up} as inputs.

Scenario 1: If the node y is a node with leaf, we consider z = $AuthAtt_y$. If $z \in E \frown Z$, $\mathrm{VerificationNode}(SigV, GL_{\mathrm{key}}, y) = \frac{k(\overline{E_z, K_y})}{k(\overline{E_{z'}, K_{y'}})}$

$$= \frac{k(\mathrm{m}^{\delta\beta/r} J_1(z)^{(r_z + t_z)/r}, \mathrm{m}^{upol_y(0)})}{k(\mathrm{m}^{(r_z + t_{z/r})}, J_1(AuthAtt_y)^{upol_{y(0)}})} \qquad (15)$$

$$= k(\mathrm{m}, \mathrm{m})^{\delta\beta/r.upol_{y(0)}}$$

If $z \in E \frown Z$, $\mathrm{VerificationNode}(SigV, GL_{\mathrm{key}}, y) = \frac{k(\overline{E_z, K_y})}{k(\overline{E_{z'}, K_{y'}})}$

$$= \frac{k(J_1(z)^{(r_{z/r})}, \mathrm{m}^{upol_y(0)}}{k(\mathrm{m}^{(t_{z/r})}, J_1(AuthAtt_y)^{upol_{y(0)}})} \qquad (16)$$

$$= 1$$

Scenario 2: If the node y is a node with leaf, $\mathrm{VerificationNode}(SigV, GL_{\mathrm{key}}, y)$ can be written as: for every node n which are the child nodes of y, it runs the $\mathrm{VerificationNode}(SigV, GL_{\mathrm{key}}, y)$ function and saves the output in P_n. Further, we assume that E_y be the random sized set of K_y children nodes n and hence, $P_n \neq$

$\perp$. If there is no such set present, it means that the node fails to satisfy and hence returns $\perp$. Else, it calculates and produces the result as follows:

$$P_y = \prod_{n \in E_y} I_n^{\Delta_{i.E_y(0)}}$$

$$= \prod_{n \in E_y} k(\mathrm{m}, \mathrm{m})^{\delta\beta/r.upol_{parent(n)}(index(n))^{\Delta i.E'_y}}$$

$$= \prod_{n \in E_y} k(\mathrm{m}, \mathrm{m})^{\delta\beta/r.(upol_{y(i).})^{\Delta i.E'_y}}$$

$$= k(\mathrm{m}, \mathrm{m})^{\delta\beta/r.pol_{y(0)}} \tag{17}$$

Upon the cipher-text update, the proposed system undergoes two theorems: one for traceability and another for key stability checking as discussed in[39].

6. ANALYSIS OF SECURITY:

To achieve higher level of security, we consider quantum cryptography method with one time pad mechanism. Hence, the proposed HQCP-ABE method is highly secure to establish data communication in fog computing. The security properties of the proposed method can be analysed as follows:

(i) **Data privacy:** Initially, the data is encrypted using access policy tree AP_t and cipher-text update policy, PC_{up} in order to guarantee the data privacy against the users who cannot contain a set of attributes which satisfies the stated access policy. The fog node involves in encryption calculations for users during encryption but it could not access the data because of the absence of secret key. The attribute sets fail to satisfy the stated access policy for the cipher-text during decryption and therefore fog nodes or cloud servers are unable to retrieve the value $B = k(\mathrm{m}, \mathrm{m})^{\delta\beta t}$ so as to acquire the anticipated value $SESK$ since it will not have the knowledge about F of the user. Hence, the users whose attributes are valid and fulfil the access policy can only be able to perform decryption over the cipher-text.

(ii) **User authentication:** The proposed scheme employs attribute-based signature to realize the cipher-text update along with user authentication. Hence, an attacker who attempts to counterfeit the signature present in the cipher-text update policy in his attributes will not get satisfied. Assume G as an attacker who attempts queries namely $upol_{J_1}$, $upol_{J_2}$, $upol_0$, $upol_{J_1}$, $upol_e$ and $upol_{t\mathrm{h}}$ to the arbitrary oracles J_1, J_2, re-distributable key generating oracle, signature oracle and secret keys generating oracle. They together make a victorious counterfeit in opposition the proposed method.

(iii) **Fine grained access control:** This enables flexibility to specify various access rights for each user. To make use of this feature, we employ HQCP-ABE method that uses quantum cryptography method. During the stage of encryption, the data owner is allowed to use a flexible access policy method and encrypt the data using quantum cryptography and re-distribute the cipher-text to the cloud servers. Particularly, the access tree which uses the access policy of the quantum encrypted data bears intricate processes that contain both OR and AND gates that are capable of representing any attribute set desired. Hence, the proposed scheme could be able to achieve fine grained access control.

(iv) **Resistance to Collusion:** The data which are not individually accessed are integrated by the users for accessing the data. They integrate re-distributable keys and secret keys. In this proposed method, the authorized attribute produces secret keys for various users. If more than two users who have dissimilar attribute merge with one another to fulfil the access policy, they are unable to calculate $G = k(\mathrm{m}, \mathrm{m})^{\delta\beta e}$ in the re-distributable decryption stage. Hence, the proposed method proves that it is resistant to collusion.

7. RESULTS AND DISCUSSIONS:

We have considered the existing approaches such as CP-ABE and Proxy aided CP-ABE (PA-CPABE) to compare with the proposed model HQCP-ABE. We implement the given instantiation for the proposed HQCP-ABE so as to assess the performance of the proposed model. The proposed HQCP-ABE method does not require any secure channel for delivering the private keys during decryption process similar to the PA-CPABE method, but all other existing approaches require secure channels for distributing the private keys to their end users in order to achieve enhanced security. We demonstrated the proposed model using the simulation parameters in Table I using NS-3 simulator. The demonstrated results are shown and discussed below. In this scheme, the attributes are varied to perform encryption and decryption operations. Moreover, the IoT devices are varied for the performance metrics such as packet delivery ratio, throughput, average delay, computational overhead and control overhead. The proposed method outperforms the existing approaches namely CP-ABE and PA-CPABE and achieved better results.

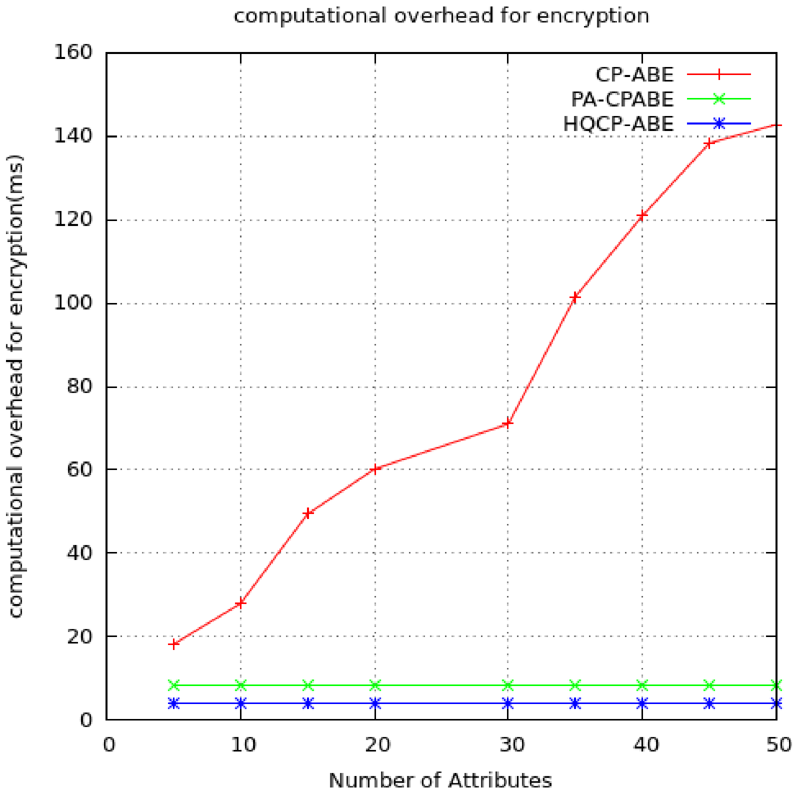


Fig. 2 Computational Overhead for Encryption

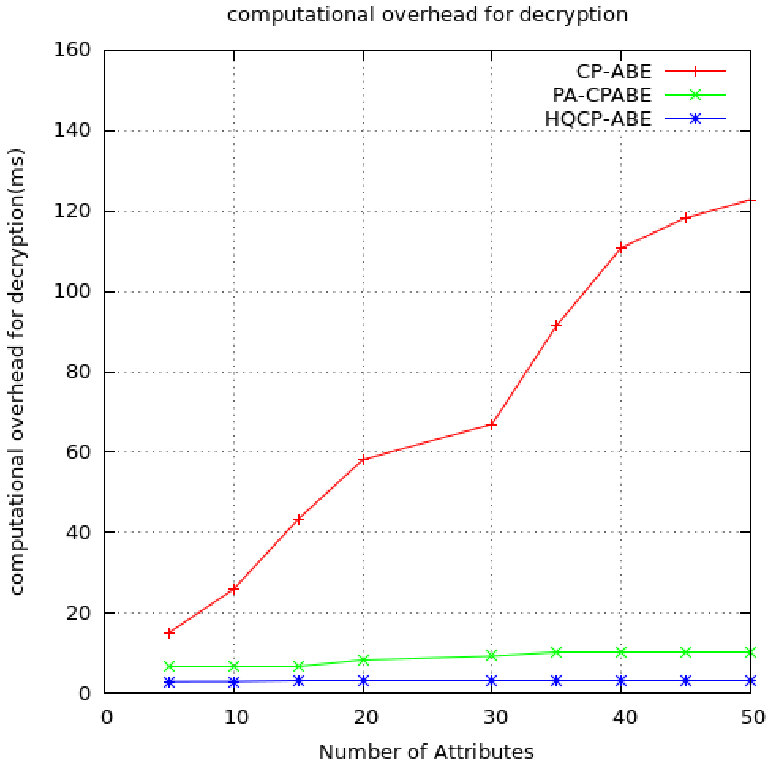


Fig.3 Computational Overhead for decryption

Figures 2 and 3 shows the results of computational overhead during encryption and decryption operations. The proposed HQCP-ABE method is compared with the existing CP-ABE and PA-CPABE methods. The graphs clearly depicts that the computational overhead for the proposed method during encryption and decryption processes are significantly lesser than the existing CP-ABE and PA-CPABE methods and hence the proposed method outperforms these two methods evidently.

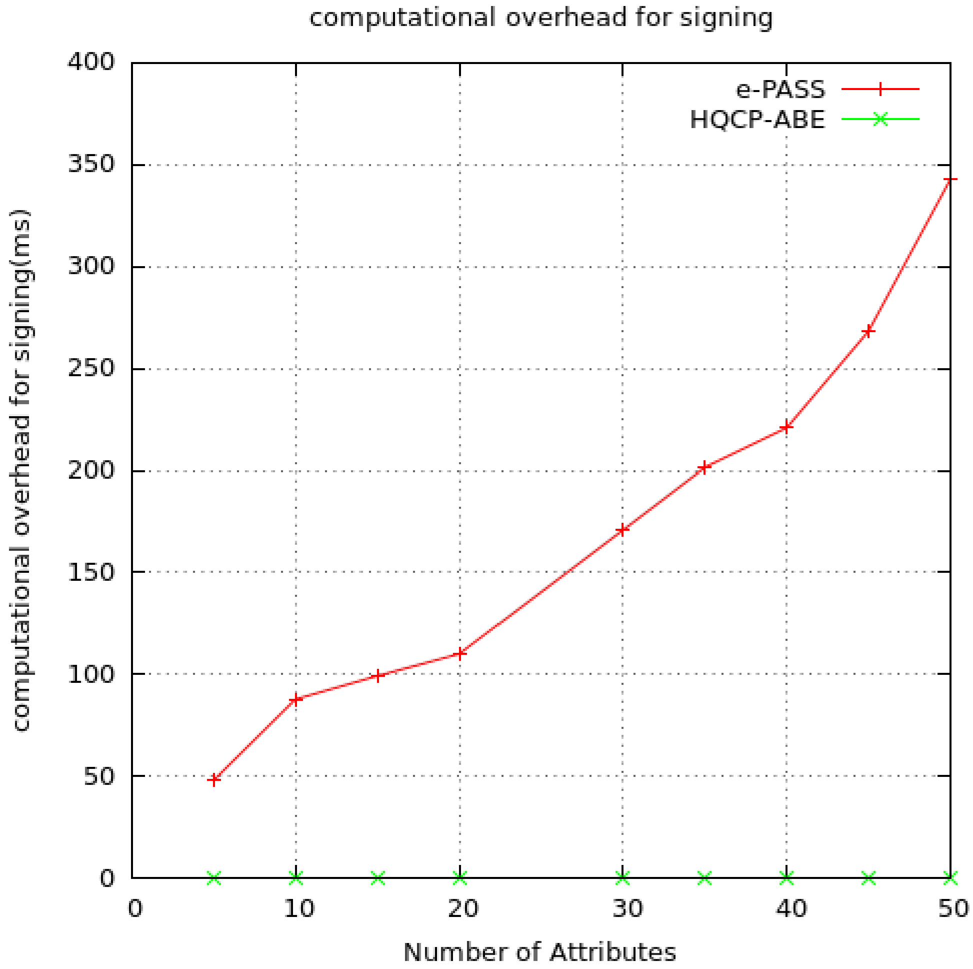


Fig 4 Number of attributes vs computation overhead for signing

Figure 4 shows the comparison of the computational overhead for signing operation. The time complexity on the cipher-text update users primarily alludes to the signature algorithms discussed in [38] is shown in the Fig 4. The proposed approach gains a constant performance while compared that of the scheme in [38] with the linearly increasing efficiency by re-distributing large number of computations to the fog nodes. Hence, this method can be utilized for resource limited IoT devices to carry out the signing operations.

Conclusions:

IoT based fog environments handle massive amount of data which requires rapid data analysis and security. Specifically, sensitive information has to be highly secured in order to ensure user privacy. Since data leakage has been a major issue among communication environments, choosing appropriate solutions that guarantee data security is vital. To overcome the challenges in the existing approaches and to ensure data security and user privacy, we present a hybrid algorithm, HQCP-ABE which is a combination of three effective techniques namely CP-ABE, cipher-text update and quantum key cryptography with OTP mechanism. The presented hybrid algorithm avoids costly pairing that occurs during decryption operation. The proposed technique effectively performs encryption, decryption and user authorization operations. The results transparently indicates that the proposed HQCP-ABE method remarkably outperforms the other two methods in terms of increased throughput, reduced packet loss, reduced average delay, improved packet delivery ratio, reduced computational overheads and control overheads. Further, the proposed system is energy efficient as well since it consumed lesser energy during the encryption and decryption operations.

REFERENCES:

[1] S. Haller, S. Karnouskos, and C. Schroth, "The Internet of Things in an Enterprise Context," in Future Internet – FIS 2008 Lecture Notes in Computer Science Vol. 5468, 2009, pp 14-28.

[2] Stojmenovic, I.; Wen, S. The Fog computing paradigm: Scenarios and security issues. In Proceedings of the 2014 Federated Conference Computer Scienceand Information Systems (FedCSIS),Warsaw, Poland, 7–10 September 2014; pp. 1–8.

[3] Krishnamachari, L.; Estrin, D.; Wicker, S. The impact of data aggregation in wireless sensor networks. In Proceedings of the 22nd International Conference on Distributed Computing Systems Workshops, Vienna, Austria, 2–5 July 2002; pp. 575–578.

[4] Lindsay, S.; Raghavendra, C.S.; Sivalingam, K.M. Data gathering in sensor networks using the energy delay metric. In Proceedings of the 15th International Parallel & Distributed Processing, 23–27 April 2001; IEEE Computer Society: Washington, DC, USA, 2001; p. 188.

[5] Tan, H.O.; Korpeoglu, I. Power efficient data gathering and aggregation in wireless sensor networks, ACM Sigmod Record 2003, 32, 66–71.

[6] Huaqun Wang, Zhiwei Wang, Josep Domingo-Ferrer, “Anonymous and secure aggregation scheme in fog-based public cloud computing”, Article *in* Future Generation Computer Systems 78 · February 2017.

[7] Alexandre Viejo, David S´anchez, “Secure and Privacy-Preserving Orchestration and Delivery of Fog-Enabled IoT Services”, Accepted Manuscript in Elsevier to appear in Ad-hoc networks, S1570-8705(18)30549-3, https://doi.org/10.1016/j.adhoc.2018.08.002

[8] Yong Yu ; Ruonan Chen ; Huilin Li ; Yannan Li ; Aikui Tian, “Toward Data Security in Edge Intelligent IIoT”, Published in: IEEE Network , Volume: 33 , Issue: 5 , Sept.-Oct. 2019, Page(s): 20 – 26.

[9] Arwa Alrawais, “An Attribute-Based Encryption Scheme to Secure Fog Communications”, IEEE Vol.5, 2017

[10] Maosheng Sun, Chunpeng Ge, Liming Fang, Jiandong Wang, “A proxy broadcast re-encryption for cloud data sharing”, Springer Article *in* Multimedia Tools and Applications · February 2017.

[11] Ahmed A. ABD El-Latif, Bassem ABD-El-Atty, M. Shamim Hossain, Samir Elmougy AND Ahmed Ghoneim, “Secure Quantum Cryptography Protocol for Fog Cloud Internet of Things”, IEEE Special Section On Recent Advances In Cloud Radio Access Networks, Volume 6, 2018.

[12] Qinlong Huang ; Yixian Yang ; Licheng Wang, “Secure Data Access Control With Ciphertext Update and Computation Outsourcing in Fog Computing for Internet of Things”, Published in: IEEE Access, Volume: 5, Page(s): 12941 – 12950, 14 July 2017.

[13] RongXing Lu, “A Lightweight Privacy-Preserving Data Aggregation Scheme for Fog Computing-Enhanced IoT”, IEEE Vol.5, 2017

[14] Kanghyo Lee ; Donghyun Kim ; Dongsoo Ha ; Ubaidullah Rajput ; Heekuck Oh, “On security and privacy issues of fog computing supported Internet of Things environment”, Published in: 2015 6th International Conference on the Network of the Future (NOF), 30 Sept.-2 Oct. 2015

[15] Amit Sahai, Brent Waters, “Fuzzy Identity-Based Encryption”, Annual International Conference on the Theory and Applications of Cryptographic Techniques, Advances in Cryptology – Eurocrypt 2005 pp 457-473.

[16] Allison Lewko, Brent Waters, “Decentralizing Attribute-Based Encryption”, Annual International Conference on the Theory and Applications of Cryptographic Techniques, Advances in Cryptology – Eurocrypt 2011 pp 568-588.

[17] Vipul Goyal, Amit Sahai, Omkant Pandey and Brent Waters, “Attribute-based encryption for fine-grained access control of encrypted data”, Proceedings of the 13th ACM Conference on Computer and Communications Security, CCS 2006.

[18] John Bethencourt ; Amit Sahai ; Brent Waters, “Ciphertext-Policy Attribute-Based Encryption”, Published in: 2007 IEEE Symposium on Security and Privacy (SP '07), 20-23 May 2007.

[19] Sushmita Ruj ; Amiya Nayak ; Ivan Stojmenovic,” Distributed Fine-Grained Access Control in Wireless Sensor Networks”, Published in: 2011 IEEE International Parallel & Distributed Processing Symposium, 16-20 May 2011

[20] Chunqiang Hu ; Hongjuan Li ; Yan Huo ; Tao Xiang ; Xiaofeng Liao, “Secure and Efficient Data Communication Protocol for Wireless Body Area Networks”, IEEE Transactions on Multi-Scale Computing Systems, Volume: 2 , Issue: 2 , April-June 1 2016.

[21] Yinhao Jiang, Willy Susilo, Yi Mu, “Ciphertext-policy attribute-based encryption against key-delegation abuse in fog computing”, Future Generation Computer Systems Volume 78, Part 2, January 2018, Pages 720-729

[22] Shucheng Yu ; Kui Ren ; Wenjing Lou, “FDAC: Toward Fine-Grained Distributed Data Access Control in Wireless Sensor Networks”, IEEE Transactions on Parallel and Distributed Systems , Volume: 22 , Issue: 4 , April 2011.

[23] Matthew Green, Brent Waters and Susan Hohenberger, “Outsourcing the decryption of ABE ciphertexts”, Conference: Proceedings of the 20th USENIX conference on Security, August 2011

[24] Junzuo Lai, Robert H. Deng, Chaowen Guan, and Jian Weng, “Attribute-Based Encryption With Verifiable Outsourced Decryption”, Published in IEEE Transactions on Information Forensics and Security, Aug. 2013, (Vol: 8, Issue: 8, pp. 1343-1354.

[25] Liming Fang, Willy Susilo and Jiandong Wang, “Anonymous Conditional Proxy Re-encryption without Random Oracle”, Springer, International Conference on Provable Security, Part of the Lecture Notes in Computer Science book series (LNCS, volume 5848)

[26] LiMing Fang, JianDong Wang, ChunPeng Ge and YongJun Ren, “Fuzzy conditional proxy re-encryption”, Research Paper, Science China Information Sciences volume 56, pages1–13(2013)

[27] Benoît Libert and Damien Vergnaud, “Unidirectional Chosen-Ciphertext Secure Proxy Re-encryption”, International Workshop on Public Key Cryptography, Public Key Cryptography – PKC 2008 pp 360-379.

[28] Osama A.Khashan, “ Hybrid Lightweight Proxy Re-encryption Scheme for Secure Fog-to-Things Environment”, IEEE Vol. XX, 2020

[29] D.P. DiVincenzo ; D.W. Leung ; B.M. Terhal, “Quantum data hiding”, Published in: IEEE Transactions on Information Theory, Volume: 48 , Issue: 3 , Mar 2002.

[30] Patrick Hayden, Graeme Smith and Debbie Leung, “Multiparty data hiding of quantum information”, Article *in* Physical Review A 71(6) · August 2004.

[31] Bassem Abd-El-Atty, Mohamed Amin and Ahmed Abd El-Latif, “New Quantum Image Cryptography Scheme with Hadamard Transformation”, Advances in Intelligent Systems and Computing · October 2017, Proceedings of the International Conference on Advanced Intelligent Systems and Informatics 2016, pp.342-352.

[32] Nan Jiang, Na Zhao and Luo Wang, “LSB Based Quantum Image Cryptography Algorithm”, Article in International Journal of Theoretical Physics 55(1) · April 2015.

[33] Keye Martin, “Steganographic Communication with Quantum Information”, International Workshop on Information Hiding, pp 32-49, Part of the Lecture Notes in Computer Science book series (LNCS, volume 4567).

[34] Xin Liao, Qiao-Yan Wen, Ying Sun and Jie Zhang, “Multi-party covert communication with Cryptography and quantum secret sharing”, Article *in* Journal of Systems and Software 83(10):1801-1804 · October 2010.

[35] J. Gea-Banacloche, “Hiding messages in quantum data”, Article *in* Journal of Mathematical Physics 43(9) · September 2002.

[36] Bilal A. Shaw and Todd A. Brun, “Quantum Cryptography with noisy quantum channels”, Article *in* Physical Review A 83(2) · February 2011.

[37] Yuancheng Li, Pan Zhang and Rong Huang, “Lightweight Quantum Encryption for Secure Transmission of Power Data in Smart Grid”, 2169-3536 (c) 2018 IEEE, 10.1109/ACCESS.2019.2893056, IEEE Access.

[38] J. Su, D. Cao, B. Zhao, X. Wang, and I. You, “ePASS: an expressive attribute-based signature scheme with privacy and an unforgeability guarantee for

the Internet of Things," Future Generation Computer Systems, vol. 33, no. 2, pp. 11-18, 2014.

[39] Xixi Yan , Xiaohan Yuan , Qichao Zhang , And Yongli Tang, " Traceable and Weighted Attribute-Based Encryption Scheme in the Cloud Environment" VOLUME 8, 2020, IEEE access, *Digital Object Identifier 10.1109/ACCESS.2020.2975813*

Figures

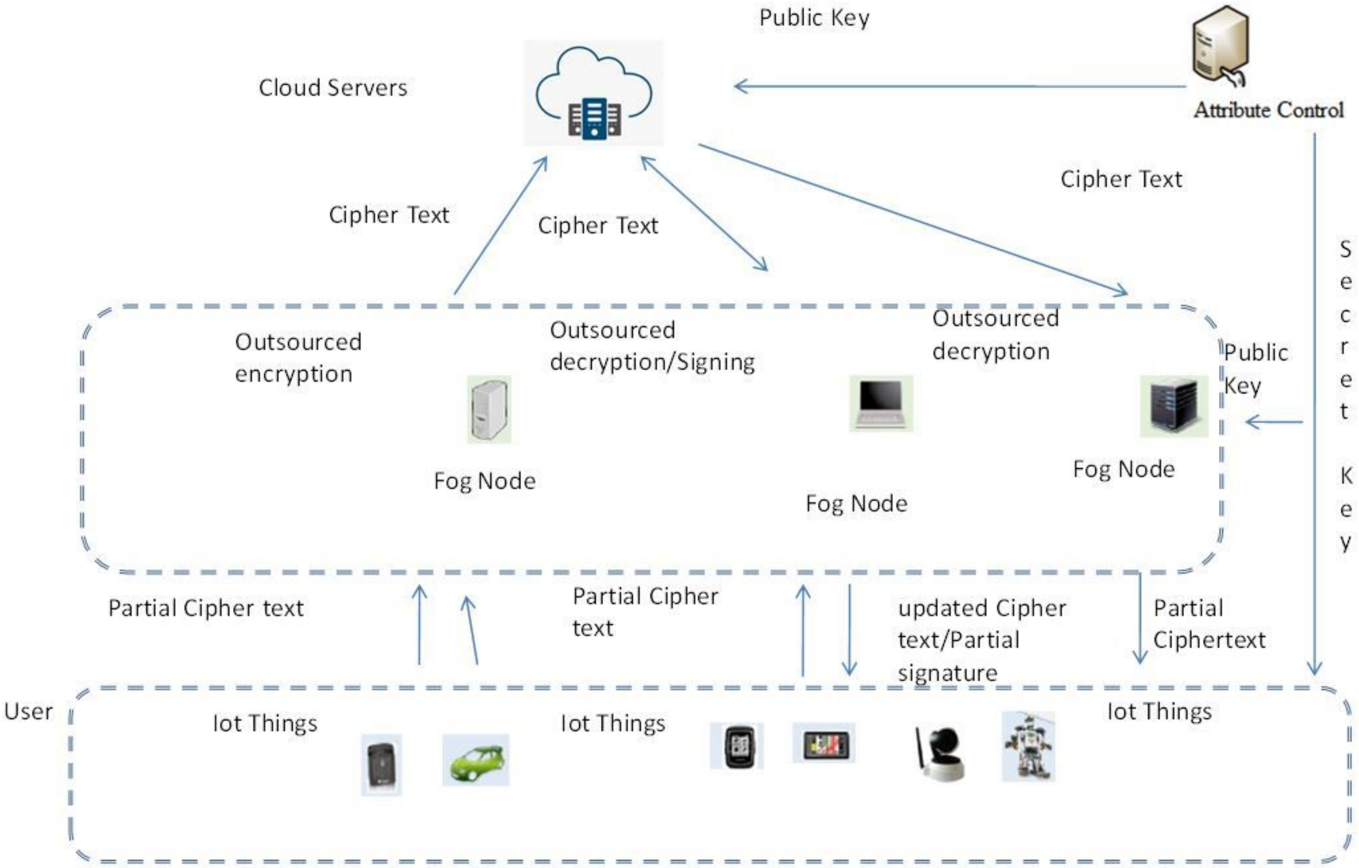


Figure 1

System Model

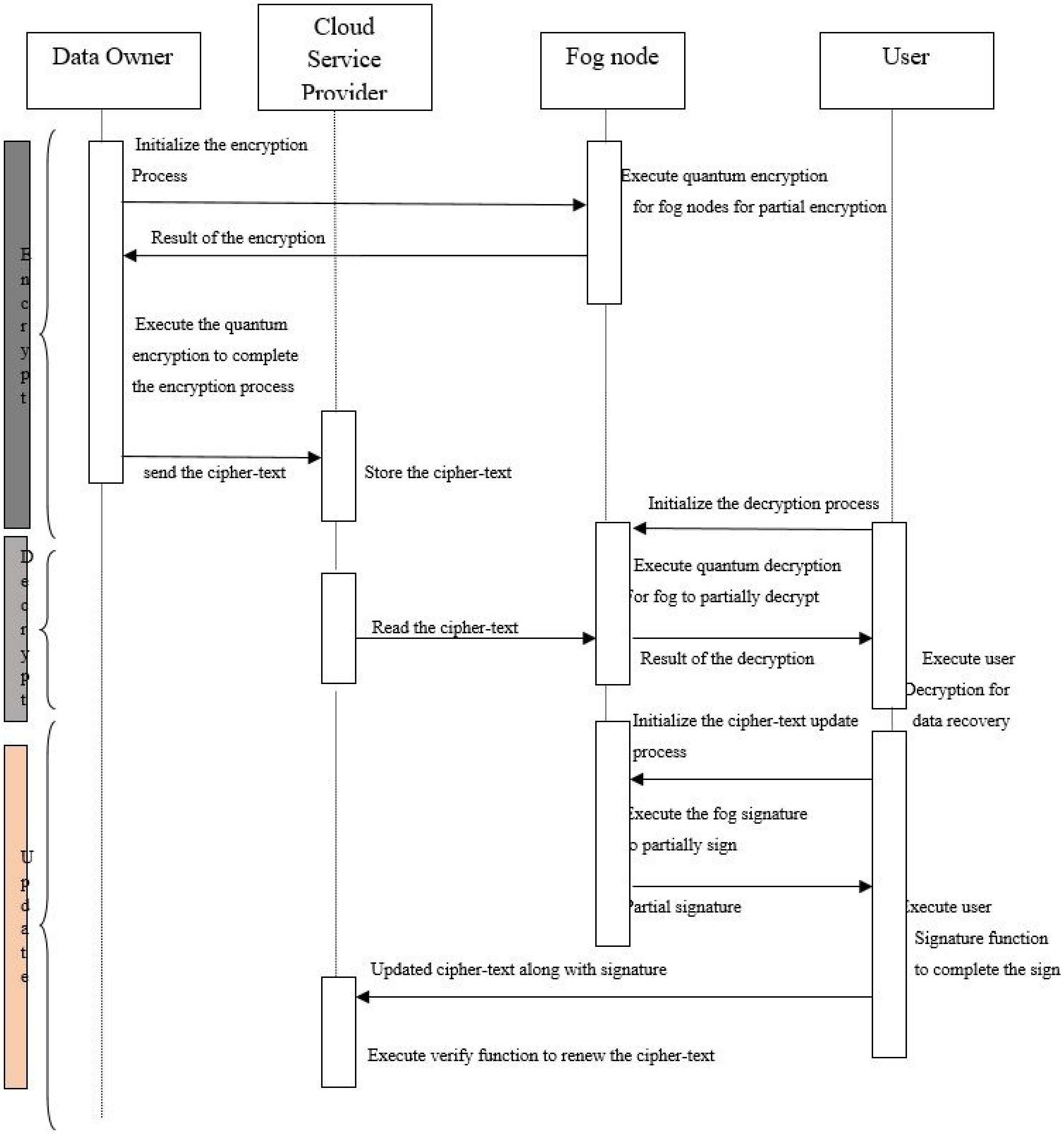

Data Owner
Cloud Service Provider
Fog node
User
Encrypt
Decrypt
Update
Initialize the encryption Process
Execute quantum encryption for fog nodes for partial encryption
Result of the encryption
Execute the quantum encryption to complete the encryption process
send the cipher-text
Store the cipher-text
Initialize the decryption process
Execute quantum decryption for fog to partially decrypt
Read the cipher-text
Result of the decryption
Execute user Decryption for data recovery
Initialize the cipher-text update process
Execute the fog signature to partially sign
Partial signature
Execute user Signature function to complete the sign
Updated cipher-text along with signature
Execute verify function to renew the cipher-text

Figure 2

Entire Workflow of HQCP-ABE method

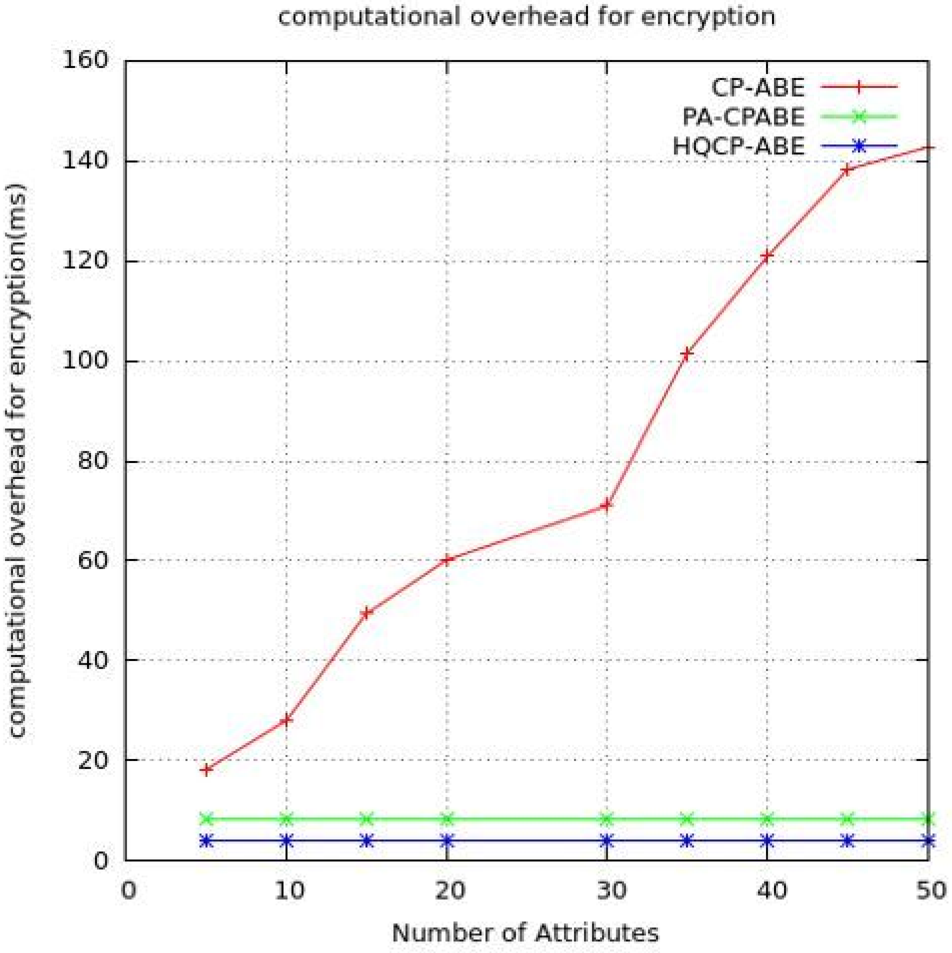


Figure 3

Computational Overhead for Encryption

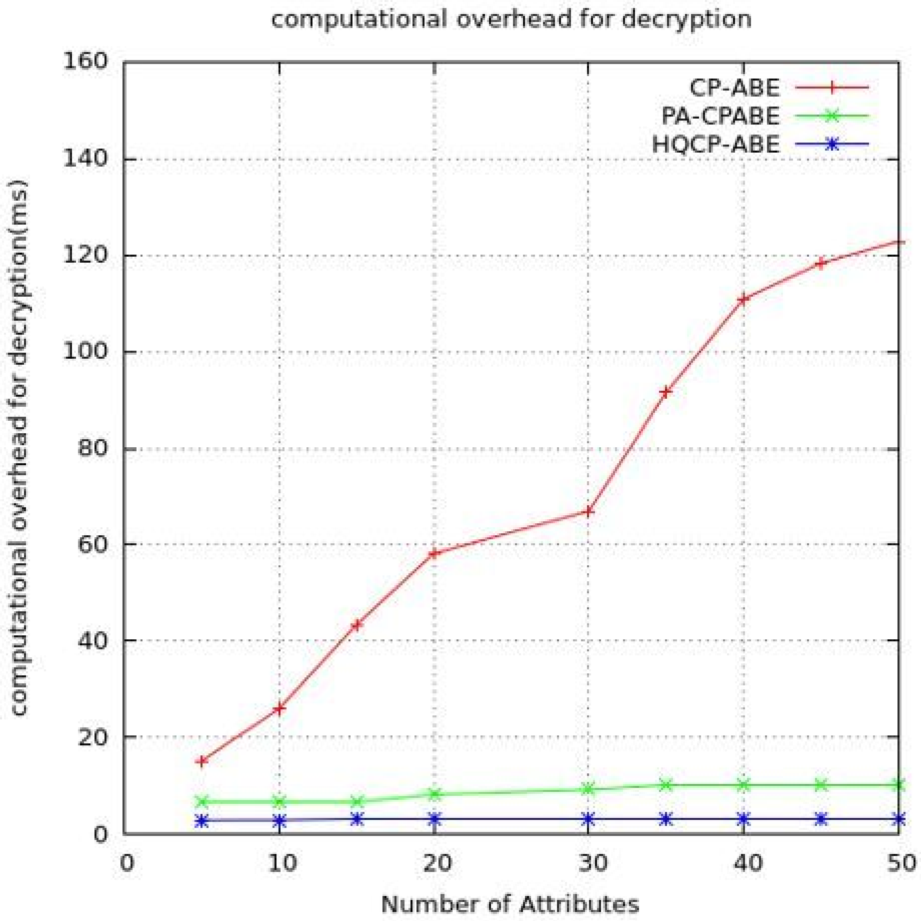


Figure 4

Computational Overhead for decryption

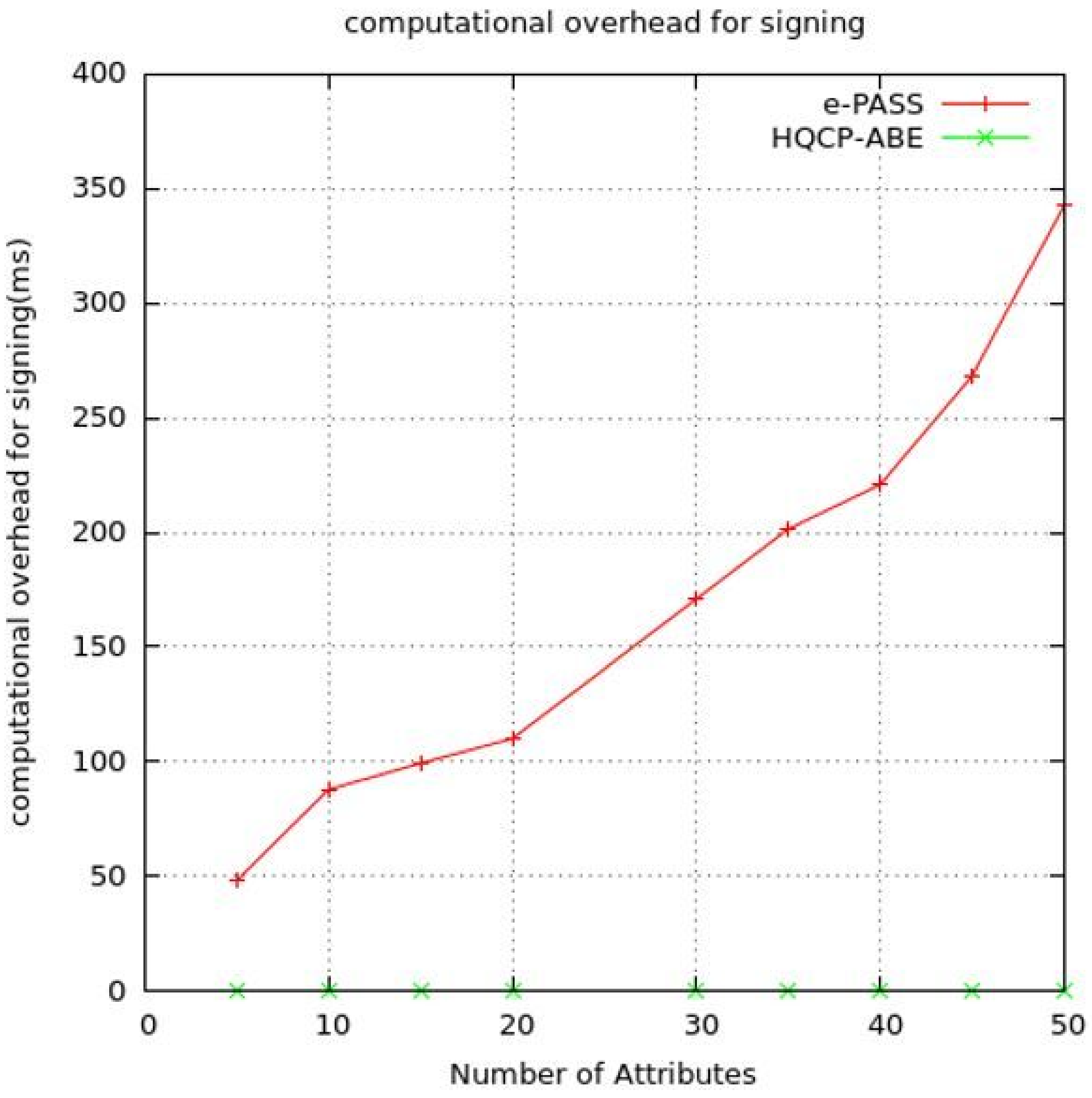


Figure 5

Number of attributes vs computation overhead for signing

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- GraphicalAbstract.docx