A scalable and secure attribute-based access control method in Internet of Things with efficient revocation

Safaa Majid Fakhry Al sherify

Ferdowsi university of Mashhad, Iran

safaacomp22@yahoo.com

Seyed Amin Hosseini Seno

Ferdowsi University of Mashhad, Iran

hosseini@um.ac.ir

DOI :http://dx.doi.org/10.31642/JoKMC/2018/070102

Received Sep. 15, 2018. Accepted for publication Mar. 16, 2020

Abstract- *The Internet of Things (IoT) is an emerging network paradigm that understands the Because IoT is connections between all things and is the foundation of the intelligent community. The pervasive, always about everyday life or daily work of the user, privacy and security are important. In other complex and heterogeneous properties of the IoT make its security issues very challenging. words, how to achieve data, while data privacy is a challenging task. In addition, resource constraints In spite of the attribute based of node creates a lightweight requirement for IoT security mechanisms. encryption like (CP-ABE), can provide access control to data by allowing specific users who their This paper provides a method of a attributes in accordance with the policy of access to decoded text. scalable and secure access control based on attribute with effective revocation based on allow users to access and as well as the possibility of expiring some users' access. Security and efficiency evaluations show that the proposed scheme can achieve the desired security goals, while keeping comparable computation overhead.*

Keywords- Internet of Things, Security, Privacy, ABE, Access Control, Attribute-Based Encryption, CP-ABE.

1. Introduction

In recent years, advances in information technology have accelerated the development of the virtual world. The internet of things is on the rise day-by-day, and momentarily, its role becomes more intense in our lives. Today, most of the end nodes on the internet are people who use smartphones, tablets, laptops, and computers, but it does not take long for things to disturb this balance, and the number of things connected to the internet is surpasses the people who are connected to the internet.

Gartner Research Institute predicts that by the year 2020 around 26 billion different devices will be connected to the internet worldwide [1]. Cisco company has also predicted that internet of things technology will connect about 50 billion devices to the internet with specific IP addresses by the year 2020. This means that an average of 6.5 devices will be connected to the internet for per person. The internet of things uses internet protocol version 6 (IPv6), which provides many benefits to the internet protocol version 4 [2].

In the widespread use of internet of things such as smart city, data ate usually collected from a variety of sources owned by different executive domains (such as smartphones, public or private transport providers). The data set may be outside knowledge of users and data transfer is performed in simple text. Since wide distributed data is shared between different departments and allowed users have access to it, if security does not exist, it may cause serious problems and these data may be even used to damage the owner of data. Another example is the internet of things application unit, which is a medical and health monitor. Normally, data

collected by body sensors of an adult patient where the patient should always be sent to the server of the medical center or hospital and the data will only be accessible to specific doctors because of body data is one of the most susceptible data. If these data are sent in simple texts or there is no proper control, the privacy of individuals may be violated, leading to serious consequences. In addition, the communication method of broadcasting wireless messages on the internet of things is vulnerable to eavesdropping [3]. Similar to the traditional networks (wired or wireless) , the data security on the internet of things include confidentiality, validity and privacy. since data is always be sent in the form of broadcasting communication on the internet of things, storage and dynamic sharing through distributed or heterogeneous networks, to encript and prevent the access to unauthorized content, it is very important getting through the encryption text (encrypted) based on the control mechanism.

Data privacy is maintained by using the encryption and its methods. Encryption keys are necessary for decryption data and must be distributed in a selected method so that users of authorized data can apply separate functions to the data. In order to manage a large number of devices that produce trickles data, a flexible and light access control solution is adopted. An appropriate approach recently performed based on the attribute encryption is access control will be flexible and light [4]. Encryption system based on the ABE feature has the nature that it enables each user to decrypt the encrypted text by creating essential features, which is very suitable for controlling the access based on the encryption text and release encryption [3].

Sometimes it is possible that users who were allowed before is excluded from the list of authorized people, so their access to data should also be taken. Due to this, it is required that dynamic system should update access by changing the list of users. In order to protect the privacy of data and confidentiality in the internet of things, an attribute-based light design is essential. Less has been talked in texts about how to expire users and how to control access in writing and reading modes, in this paper, we try to create a scalable and secure encryption method to control the access of individuals to information based on their permission and the possibility of expiring the access of some users. In this method the encrypted new text that is sent to the user, the signature of the user who is allowed to write is sent, too, and it done by outsourcing part the encryption and decryption process to a fog server that is somewhat trusted. Two-parts revocation procedures will also be used to expire the access of individuals.

The rest of the paper is organized as follows: In Section 2, related works is describes. Section 3 is described proposed approach. Section 4 a Security analysis of proposed method. Simulation platform is presented in Section 5. The evaluation of the proposed approach is presented in Section 6. Finally, Section 7 concludes the paper.

2. Related works

In this section, we study the works done on the encryption design based on the light feature on the internet of things, also we categorize these papers according to the algorithm used in them. 1. Encryption CP-ABE 2. KP-ABE

2.1.Works done by the encryption CP-ABE

Pang et.al. [4] Applied a pre-computation method to the encryption algorithm CP-ABE to allow overcoming the computational cost of encryption measured by the complexity of the access policy and the number of features. The proposed method can be considered optimally in the size of encryption algorithm to reduce practical problems in implementing CP-ABE on devices with limited resources. The authors shown that with this method, the energy storage advantage will be in terms of computational cost. If the storage requirements of the method become an important concern, a hybrid approach can change the pre-computation and demanded computations to overcome this problem.

Toatti et.al. [5] Tried to help the heterogeneous nature of the internet of things to use the CP-ABE scheme in the internet of things environment, their main idea is to paste loads of devices with very limited resources on unrestricted resources generated by heavy operations in the CP-ABE design of unconstrained neighboring nodes. The basic idea for exponential computation in known to other trusted neighboring devices, called assistant nodes. When a device with limited source wants to encrypt a message, it is searching trusted unlimited nodes in its neighboring, and this will expose them to costly operation. Moreover because of the initial encryption CP-ABE , the load changes from devices with limited resources to devices with unlimited resources.

Toatti et.al. [6] Addressed an important issue, which is abandonment of the feature for a feature based on encryption schemes. In particular, they considered practical application scenarios in which the owner of feature knows beforehand the start time and the duration of all credits validity periods

for the specifications, and propose a design that will support the cancellation of feature. An appropriate feature of their proposed scheme is that there is no surplus nature in the network, such as proxies, and we do not need data which is re-encrypted to cancel it. Their proposed solution here produces a zero delay and a minimum of hidden key sectors.

Zikon et.al. [7] Showed that the prototype pf the ABE implementation of the project provides the results that can be used in the descriptive scenario. By increasing the processor power on mobile devices and implementing encryption functions may be involved in the hardware layer, the results can be definitely upgraded in the future. Introduction of server functionality can support the use of ABE in mobile cloud data scenarios. The question solution of how key distribution, in an attitude-based encryption scenario is implemented, requires to take into consideration in the later versions of user sideways software and servant. The main provided scenario is also works to launch only the mobile users. The storage service extensions like Dropbox, and additional components added to open distributions such as Own-cloud, will be included in the future prototype. Describing functions of access records and fully extended compartment, is the next step to be merged, as well as the introduction of dynamic features ABE such as local information are also be merged.

Hoor et.al. [8] Provided an attitude-based data sharing scheme to describe a data access control that appropriately use data sharing features. The proposed scheme has a key issuance mechanism that sets the key storages aside during the production of keys. The secret keys of users are created through secure double-sided computations, so that any key production, curiously or data storage, cannot individually acquire these personal keys. Therefore, this scheme will improve the privacy of information and reliability in data sharing systems against all system administrators as well as outsourcing without corresponding (and enough) confidentiality features. This scheme can terminate users immediately in any feature set, if necessary, and can use scalable access control that is provided by attitude-based encryption with the encryption message policy. Hence, this method can provide the better access control to data in data sharing systems. They showed that this method is an effective and scalable way to manage data securely in the data sharing system.

Jiang et.al. [9] For the first time in articles, tackled the issue of key assignment in attitude-based encryption systems with coding message policy (CP-ABE). They introduced a new mechanism to introduced CP-ABE designs that provide enough protection against these key granted abuse problems. They set security requirements for these features the and then created a CP-ABE design that meets security requirements. They also provided an application of their own design on an exploratory CP-ABE that in this part, abusive users, means the users who gave their keys to others, could be tracked.

Jang et.al. [10] Provided for the first systems of general sharing of attitude-based information, based on the CP-ABE combination mechanisms and symmetric encryption schemes. Then, they described a CP-ABE design that had stable computational costs of the encrypted messages with constant size. This implies that a design CP-ABE, is a safe alternative method in predictive models under the assumption of decision n-BDHE, where n represents the total number of features in general. This method can effectively utilize the access policies for AND gates with different values in features and access cards. Theoretical analysis and test results indicate that the proposed scheme is highly suitable for data sharing in mobile cloud networks.

Jang et.al [11] introduced a new technique called decryption after matching, in which a matching step is provided before the decryption step. This technique works by calculating specific sections in the text message used to run the test, if private keys are considered to be hidden access policies and consistent in not decrypted coded messages. For fast decryption issue, key coding sections and their special features are created which allow the accumulation of these pairing during decryption. They provide an unknown ABE structure and then create security improvements based on the unforgiving timed signs. In this presented structure, the computational costs of the tests are lower than the decryption costs according to the features, and it requires only a small number of pairing. Official security analyzes and functional comparisons show that proposed solutions can simultaneously guarantee privacy and provide decryption efficiency for storing outsourced data.

Han et.al. [12] Provided a new design CP-ABE that can protect user attribute values against AA by using conscious transfer method 1-out-of-n. In addition, it uses ABF to protect the feature of the type of access policy in the encrypted text. Finally, security and performance evaluations show that the proposed scheme can achieve the desired security objectives, while maintaining comparable computing overhead. The paper plan has six algorithms as follows: setting up,

producing key, creating ABF, encryption, checking ABF, and decryption.

Brug et.al. [13] introduced two methods for how to use CP-ABE on systems with limited resources. The feasibility of two systems is evaluated based on the experimental results ABE in a resource-limited sensor. The most appropriate solution is determined by the sensor computing power, the maximum policy length and the time required for sensor application.

2.2. Works done by encryption KP-ABE

Avalaha et.al. [3] Proposed a light ABE design for limited resource unit in internet of things based applications to provide secure communications and access control of encrypted message. Considering the light advantage and the primary mode KP-ABE, both lightweight and ABE are available in the proposed design. Comparative analyzes have been made on existing design KP-ABE and designs CP-ABE to indicate that the proposed scheme is a lightweight, which is not only has no data overhead but a small computational overhead. In addition, its restrictions and flexibility, scalability and applications with multiple licenses are also discussed. Totally, the proposed design was a lightweight KP-ABE design and it is suitable for the internet of things unit with limited resources.

Singh et.al. [14] Analyzed the possibility of CP/KP-ABE to provide communication security for internet of things based systems based on the architecture Pub-Sub. As a part of this issue, they suggested that secure MQTT (SMQTT, SMQTTSn) protocols be implemented with a secure publication order (SPublish) that emits encrypted data based on the KP/CP-ABE design using lightweight ECC methods, which is done by the optimizing parameters and computational algorithms on elliptic curve. Security analysis investigated more than SMQTT based on the different attack scenarios and there is still a possibility for SMQTT for distribution architecture Pub-Sub based on a global basis. Through simulation analyzes, they indicated that application SMQTT based on the CP/KP-ABE for different requirements of internet of things (such as static/dynamic, interactive/non-interactive by PKG with devices and so on). They also showed that in theoretical and experimental point, the SMQT based on the CP/KP-ABE design works better than other methods.

Shei et.al. [15] Provided a different method of ABE that with the name of ABE declares a cancellable key policy directly with the verifiable representation of the encryption message. Their solution can be used in information sharing applications in cloud situations and it provides the possibility of following: 1) Trusted credentials to terminate users, especially by updating the termination list without interacting with the non-terminated users. 2) Third person (means the provider of information storage) to update the encrypted message text to ensure that end users cannot decrypt the texts. 3) Each auditor can confirm that a third person , on him there is no any trust, can re-read the text of messages correctly or not.

3. Proposed Method

To read the information of a policy tree, and for writing and signing, a policy tree is designed, and is sent to the user along with the data to be sent. To achieve the first goal (write access control), accesses should be explored separately. The proposed algorithms, so far have been like that if the specified user was allowed access to the information, he could change or update these information (write), in addition to read the existing information (read). But with passing the internet of things networks and differing the user requirements level, it seems that access controls are being considered separately. It means that if a user requests access, he will be asked to specify the level of his need, that whether he needs to read the information or wants to make a change in them. One solution is to define a policy for read and a separate policy for write, and depending on the user permission level, he can get permission for one or both of them. So, by changing this part of the basic algorithm, the write access control can be well implemented.

To achieve a method for expiring individual access after a specified time (revocation), a variety of expiration methods were investigated, that are divided into following three categories:

- Expiring an attribute
- Expiring a set of attributes
- Expiring a unique identity with a user

Since, in the basic algorithm, the user that no longer has access should be expired, then we use the third method to do this task which will be formed from the following three steps:

- Creating the modified keys
- Updating the other key of users
- Re-encryption of data

At the beginning of the access control methods, there is a stage in which the public key of each authorized user is generated, and the user can perform encryption and decryption operations with it. That’s why it is done for all users, but if

the user has expired, he should not have the ability to decrypt. Since users do not have access to the key and expired users will always be able to decrypt the encrypted data and affect the system security, we cannot perform operations such as getting the key from them, we have to change the key of the rest of authorized users and decrypt the data with these new keys again. The modified key generation process is similar to the same public key generating operation at the beginning of the proposed method. It should be noted in the proposed method that no combination of the expired user keys should be able to decrypt new encrypted data.

3.1. Create the write access control using attribute-based signature methods

In order to achieve this goal, the accesses should be explored separately. The proposed algorithms, so far have been like that if the specified user was allowed access to the information, he could change or update these information (write), in addition to read the existing information (read). But with the expansion of the internet of things and differing the user requirement level, it seems necessary to check access control separately. It means that, if a user requests access, he will be asked to specify his requirement level that whether he needs to read information or he wants to make changes in them. One solution is to define a policy for read and a separate policy for write, and depending on the user permission level, he can get permission for one or both of them. So, by changing this part of the basic algorithm, the write access control can be well implemented. For this purpose, two policies read and write are examined. According to that, in this proposed system, cloud is a server which has less security, and should not have the username and passwords for authentication. For this purpose, the attribute-based signature will be used. In this method, the signature of the user who is allowed to write will be sent with the new encrypted text that wanted to be sent to the user. This is the attitude-based signature that the user has it and it is created based on the write access. In other words, along with the access policies, the other policies will be defined to users that show the write policies and in fact, it determines the users who are allow to update the information.

The first algorithm shows the way of function of the proposed method.

Algorithm 1 - Write access control using signature-based method

```
1.Input: writePolicy, User,
     pubKey, data
2.//Encryption Phase
3.signature =
     Cpabe.encrypt(pubKey,
     writePolicy, data)
4.encryptedData=
     Cpabe.encrypt(pubKey,
     Policy, data)
5.//Decryption Phase: Read
     Access Policy
6.attKey =
     Cpabe.keygenSingle(pubKey,
     User.Attribute)
7. if  User.UserAccess== Read
     then
8.
          decryptedData=Cpabe.de
     crypt(attKey,
     encryptedData)
9.end  if
10.  //Decryption Phase: Write
     Access Policy
11.  if  User. UserAccess==
     Write then
12.   if  Cpabe.decrypt(attKey,
     signature) is successful
     then
13.
          decryptedData=Cpabe.de
     crypt(attKey,
     encryptedData)
14.   else
15.        Access Request
     Denied.
16.  end  if
17.  end  if
```

Data, is the data that will be stored on the cloud, which should be stored at the encrypted form. Since we have two policies for writing (write policy) and reading (read policy) the information in the proposed method, we use write policy for writing or changing the information, and also read policy for reading them. Each data owner who wants to store his data on the cloud, will place the read and write policies with the encrypted data, in the data. Also, for showing the accuracy of stored information, he will place his signature next to the data, for these data based on the policy tree. If a user has the permission of writing, updating or changing, he can use his attributes,to confirm the accuracy of signature and read or change data by receiving the encrypted data.

In this algorithm, first the signature will be encrypted by the help of write policy, and then the main data will be encrypted by the help of access policy. (lines 2 to 4) in the decryption stage, the operation will be done based on the requested access of user, in this way that if a user requests "read", the decryption will be done regardless of the write and sign policy (lines 6 to 9). But if the user requests "write", first it will be examined that whether the user has the possibility of updating information according to the write policy or not, in the way of existing this possibility, he will be allowed for decryption, but otherwise, his request will be refused (lines 10 to 16).

3.2. Considering the possibility of expiring individual access after a specified time using two-part revocation method

For all encryption systems that support a large number of users, the private keys can be threatened or users can leave the system or being fired for the various reasons. As soon as any of these conditions occur, the keys of these users must expire and be deleted from the relevant encryption system. For this purpose, various methods are presented, in which we will refer to the following:

In some indirect expiring methods, that requires a specialized center to publish the updating of keys, periodically only unauthorized users can update their keys and decrypt their relevant messages. On the contrary, there are direct expiring methods in which a list of expired users is specified to be taken into consideration in the encryption process, .thus there is not any permission for decryption of the final encrypted text by the key of users whose name is in the list of authorized users. Even if the set of attributes associated with these keys be able to meet the policy related to the encrypted text lonely, then the permission of encrypting data should not give to the owners of these attributes whose names are in the list of expired individuals.

By comparing these two methods, we conclude that direct expiration mechanisms do not need to periodically update the keys which are implicit in the indirect method. In addition, in direct expiration methods, other users who still have permission to use the encryption system, will not be affected and they do not need to be updated or changed. But on the other hand, in the direct expiration methods, it is necessary to provide a list of users and unauthorized keys vertically, and as soon as a key that is related to an unauthorized user is removed from the encryption cycle, this list also needs to be updated. For this reason and according to the advantages and disadvantages of the two defined methods, we conclude that the use of direct mechanisms in terms of time (delay), energy and security aspects, seems to be more cost-effective and logical. Therefore, we base our research on extending and adding these types of methods of expiring the unauthorized users to the algorithm system in this paper.

Of course, it should be noted that our proposed method used the users expiration method in which the access of unauthorized users to the messages that are generated after expiring the user will be prevented,. However, the attempt on this aspect can be done to control the access of messages which have been already encrypted.

At the proposed method that presented, for encrypting the data, a list of expired users by the name of revocation list is needed in addition to the data that will be encrypted. in other word, encryption function needs 3 inputs: list of expired users, access policy and data. Each user will be identified by a pair of information (ID, attribute set), which represents the user's ID and a set of attributes for this user. Now, due to that a user will be able to decrypt a message, he must meet the following phrase by using the information (attribute set and index) he has:

$(\text{User.id} \notin \text{RevocationList}) \wedge (\text{User.AttributeSet satisfies Policy})$

This means that, in addition to that the attribute set of A user should meet the access policy conditions, the ID of this user which is user.id, must not exist in the list of unauthorized and expired users (revocation list). Thus, in the presented method, the possibility of expiring users is provided at the moment of encrypting new message, and only the authorized users (not the expired users whose ID are exist in the revocation list) will have the permission to access the new encrypted messages. It is only necessary to put the user ID in the expired users list as soon as the user access permission to the messages has foreclosed, and we will update this list for new messages.

It should be noted that because the revocation list is the only set of IDs or numbers, if this list is disclosed by the attackers, no special information will be provided to them, and it will not provide any information of users for the attackers. Because this list is joined to policy in the form of encrypted, it is not possible to change or remove it by the individuals who have the required qualifications (only the owners of this data have this permission).

The second algorithm shows the way of expiring users.

Algorithm 2 - Expired users algorithm

```
1. Input: Policy, User, data,
   RevocationList, pubKey
2. //Encryption Phase
3. newPolicy=UpdatePolicywithRe
   vocatedUsers(RevocationList)
4. // (User.id ϵ
   RevocationList) Λ
   (User.AttributeSet satisfies
   Policy)
5. encryptedData=
   Cpabe.encrypt(pubKey,
   newPolicy, data)
6. //Decryption Phase
1.  attKey =
   Cpabe.keygenSingle(pubKey,
   User.AttributeSet)
2.
   decryptedData=Cpabe.decrypt(
   attKey, encryptedData)
```

In this algorithm, a new policy creates a new access policy with the help of adding the non-expired users list (means the users who are not in the revocation list) (lines 2 to 4). then data encryption is performed using a new policy, and the rest of the encryption routine will be done as usual (lines 4 to 7).

4- Security analysis

The fundamental challenge of the proposed method in terms of security, is the collusion attack between the users and the expired ones, thus we have secured our method against these attacks. In the proposed method by using the list of expired users, which is continuously updated, it will eliminate the interference between users completely .

Another attack that may be performed on the encrypted data, is the middle-man-attack, in which the data are encrypted by the policy tree and only those users can encrypt the information that their attributes can meet the policy tree, but the attackers do not have these attributes and the possibility to decrypt.

5. Simulation platform

In order to implement the proposed algorithm and examine the differences, a topology is considered for the network. In this matching, a processing element named cloud, is used which its attributes are showed in table (1). This element is placed at the level zero of matching, because in the IFogSim simulator, there is a hierarchical structure for the things with different processing power. Also at the level 1, there is an attribute authority server. At the proposed structure, this server is considered as the fog type, and with its relatively high processing power can perform tasks such as checking access to data as well as attitude-based policy creating and data encryption operations. Also the users are at the leaf level (level 2) that are directly connected to the fog, and send their requests, which are based on the access to data (whether write access or read access), Moreover, fog sends data that is encrypted based on the attitude-based policy, to the users. So a user whose attributes can handle the policy associated with data, can achieve encrypted data and finish the decryption operation correctly.

Device	Mips	RAM	Up Bw	Down Bw	Rate/ Mips	Level
Cloud	10000	8000	100	1000	0.25	0
Fog	1000	1000	1000	10000	0.0	1
User1	300	500	1000	1000	0.0	2
User2	300	500	1000	1000	0.0	2
User3	300	500	1000	1000	0.0	2
User4	300	500	1000	1000	0.0	2
User5	300	500	1000	1000	0.0	2
User6	300	500	1000	1000	0.0	2

Table 1- Devices information

Mips: million instructions in second (processing power)

UpBW: upward bandwidth in the hierarchical structure

DownBW: downward bandwidth

Rate/Mips: the amount of calculated cost for each million used infrastructures

Level: rolled level in the tree structure

In the designed matching, six users have been included to consider different user states (figure 1). Half of the users, request reading information (read access) and the other half, request changing or writing information (write access) that is mentioned below:

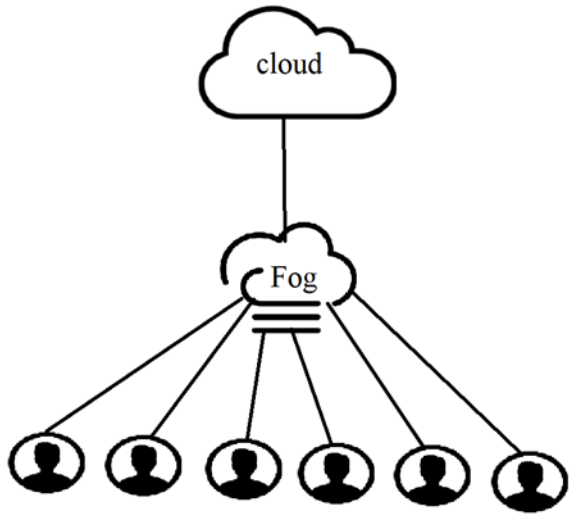


Figure 1- Network topology

1) The first user is the one who requests to read the information. This user is not in the revocation list (R list), and he is allowed to receive the information. Because the information has been sent to this user, the attitudes that are available to the user are able to reopen the policy associated to information and thus be succeed in reading the sent information.
2) The second user also requests to read the information and he is not in the R list, Moreover, he is not like the users whom the information is sent to. For this reason, it cannot reopen the attitude-based policy with its attitudes and will fail to decrypt the information.
3) The third user also requests to read the information but because of some reasons it is placed in R list and no longer can receive the information, so it will fail to decrypt the information.
4) The forth user is one who requests to write. This user is not in the revocation list and is allowed to receive and change the information. Because of the reason that information has been sent to this user, the attitudes which are available to user can reopen the policy associated to information and thus it can read and change the information.
5) The fifth user also requests to write the information . he is not in the R list, and he is not like users whom the information is sent to. For this reason, it cannot reopen the attitude-based policy with its attitudes and will fail to decrypt and change the information.
6) The sixth user also requests to write the information, but for some reasons he is placed in the R list and no longer can receive the information, so it will fail to decrypt the information.

5.1.Scenarios and related applications

In order to implement the proposed method in the IFogSim [16] software, one or more scenarios for simulating the network and structures and method evaluation should be presented with regard to the designed matching. So, after reviewing a few points, we will describe the scenarios and applications that used those scenarios.

- It should be noted that the delay of the connection link between fog and cloud is much greater than the connection links between fog and users (Table 2), because fog is locally placed in the user locations and their connection has less delay.
- In addition, because working cloud and utilizing its services are costly, we have been trying, to take over the average processing power tasks on the fog for minimizing the cost and delay, as far as possible.

Table 2- Delay of links

Source	Destination	Delay
User1	Fog	5.0
User2	Fog	5.0
User3	Fog	5.0
User4	Fog	5.0
User5	Fog	5.0
User6	Fog	5.0
Fog (Via Internet)	Cloud	30.0

In the first scenario where the application is also written based on it, the work process will be as follows:

1) The first step is to request access from the user, which is called *Access Request*. In this step, the user sends its request to the fog, which can be the request of write or read information. A module with the same name is placed in the related application and the output edge is also named Request.
2) The second step is an access request checking step, which is called Check Access. In this

step, the fog will examine the user requests whether they are allowed to write information or not (if the request is a type of writing), and sends the request to the cloud. A module with the same name is placed in the application and the output edge from it to the cloud is also called Reply.

3) The third step is occurred in the cloud and it is called Check Availability. In this step, the cloud responds to a request from the fog to ask whether such information exists in the cloud, and whether the specific access permission has been considered for it or not. Then, it will send its respond to the fog. A module with the same name will be designed in the application and its output edge is called Reply.
4) In the fourth step, the fog will perform several operations on a response which received from the cloud. This step is placed at the scenario with the name of AttPolicy Making & Encryption. In this step, the fog checks out what response the cloud has been sent to it. If the response contains the requested information, the fog will begin to generate the policy based on the authenticated user attitudes. Then, it will associate data to this policy and encrypt them, and will send them to the users. A module with the same will be placed in the application and the output edge from it to the user is called Encrypted Data.
5) In the last step, regarded to that, whether the user was allowed to request a write or not, and also depending on the attitudes that he has, he will be able to access information or will be unable to decrypt them. This step is called Decryption and a module with the same name is designed in the application.

The overall scheme of the scenario which is described in figure 2, is presented in the IFogSim structure.

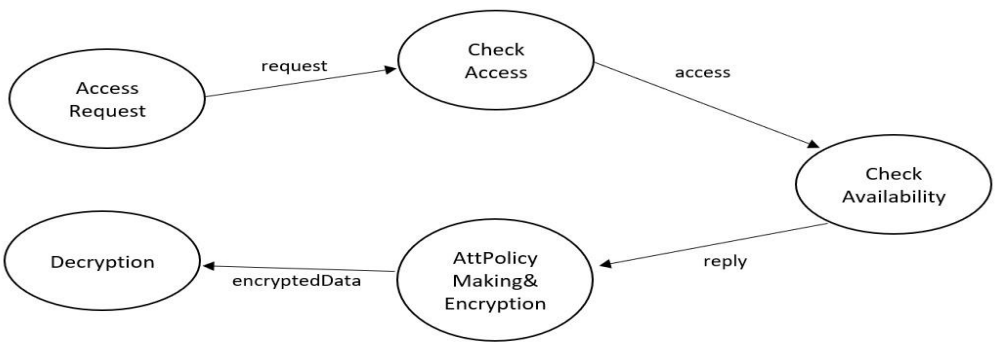
Figure 2- The overall scheme of the scenario

6. Evaluation results

To evaluate the proposed solution, the obtained results are compared with the scheme presented in [12] with the name of Han's Scheme. because the evaluation criteria are similar to this paper. For this reason, one of the most important factors which will represent the improvement of the proposed method, is the efficiency of the proposed encryption method. Among these criteria, that will be evaluated and compared, is as follows:

- Encryption time
- Decryption time
- Key generation time
- The number of used attributes
- The amount of using RAM

Figure 3, compares the efficiency of the key generation algorithm between two designs. In this experiment, we set the number of user attributes from 5 to 30. It is clear that the cost and time will be increased with the number of attributes of the both designs. Because the proposed method does not use Oblivious Transfer (OT) when generating a key. this is an access policy that can hide the attribute amounts in the key generation against the Attribute Authority (AA) [12]. So, during the key generation time, the proposed method should be less to half or even less. Han method uses Garbled Bloom Filter (GBF), and it has more encryption and decryption time. Bloom Filter (BF) is a data structure which uses the space optimally. This structure was used over the years to check the



membership of an element in a collection. Thus, if one of the corresponding bits is zero, it can definitely be said that this element is not a member of this set; otherwise, with an error probability, it can be said that it is a member of the set. This error probability is because of the existence of false positive in data. But, GBF is an improved structure that instead of using a bit, uses a string of bit and is able to decrease the false positive. For this reason, this method in addition to that, depends on the probability of interference of hash functions, and on the matching of bit string, and As you see in figure 3, its time is more than the proposed method time. The results are represented in table 3.

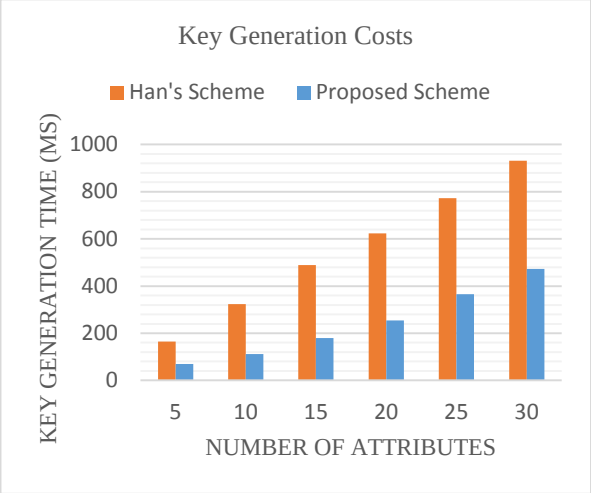


Figure 3- Comparison of key generation

Table 3- The results of key generation cost

Number of Attributes	Han's Scheme	Proposed Scheme
5	165	70
10	324	112
15	489	180
20	623	254
25	772	365
30	931	472

Figure 4, shows the comparison of encryption algorithm efficiency between the two designs. The encryption time increases with the number of attributes in the encrypted text in both designs. Both in the proposed method and Han method, the increase in the number of attributes increases the encryption time because the increase in the number of attributes, increases the creation process of access tree and encryption and also decryption, since the text should be encrypted and decrypted based on the more attributes. Han design uses GBF to protect the access policy. Of course, because it uses GBF it needs more encryption and decryption time and its encryption time is more than its proposed method time. The results are represented in table 4.

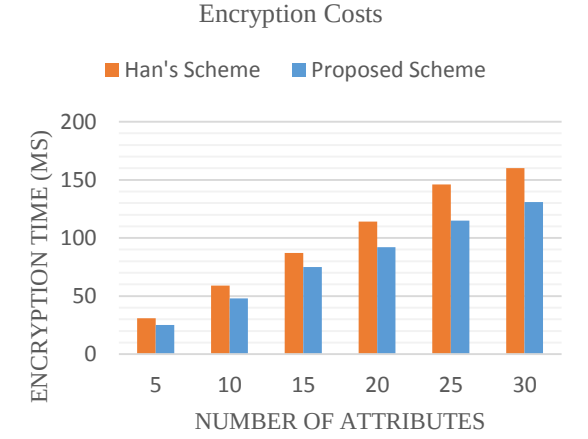


Figure 4- Comparison of the encryption time efficiency

Table 4- The results of encryption time

Number of Attributes	Han's Scheme	Proposed Scheme
5	31	25
10	59	48
15	87	75
20	114	92
25	146	115
30	160	131

Figure 5, shows the comparison of decryption algorithm efficiency between the two designs. The encryption time increases with the number of attributes in the encrypted text in both designs. Han design uses GBF to protect the access policy. Of course, because it uses GBF it needs more encryption and decryption time and its decryption time is more than its proposed method time. The results are represented in table 5.

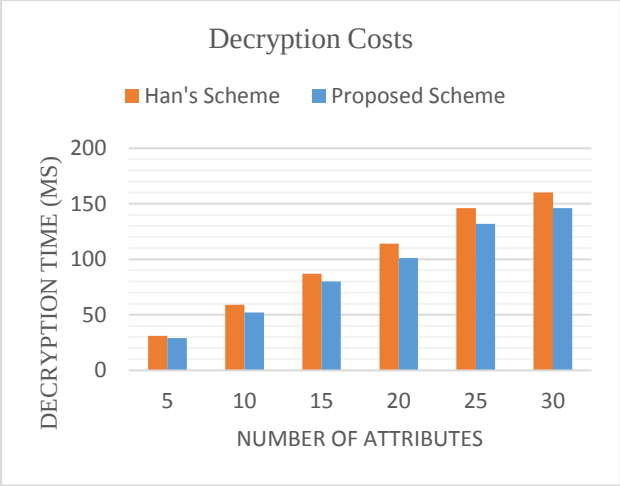


Figure 5- Comparison of the decryption time

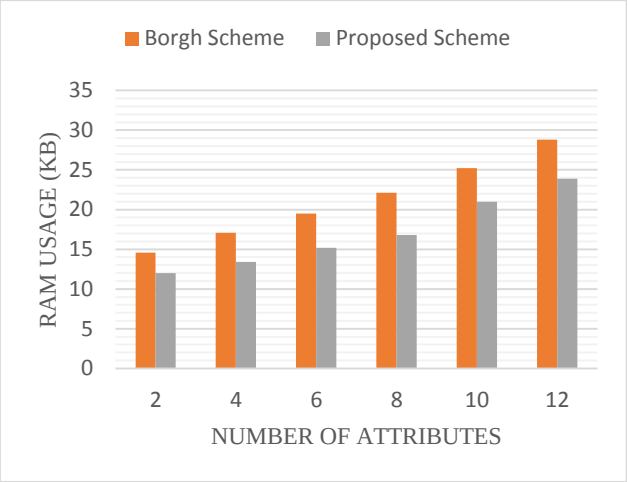


Figure 6- Comparison of using RAM

Table 5- The results of the decryption time

Number of Attributes	Han's Scheme	Proposed Scheme
5	31	29
10	59	52
15	87	80
20	114	101
25	146	132
30	160	146

Table 6- The results of using RAM

Number of Attributes	Borgh Scheme	Proposed Scheme
2	14.6	12
4	17.1	13.4
6	19.5	15.2
8	22.1	16.8
10	25.2	21
12	28.8	23.9

Figure 6, shows the comparison of using RAM between two proposed designs and Borgh [13]. The reason of why the use of RAM is less in the proposed method, is that the security level is low in the proposed method and the privacy is more being into consideration. If the security level in the system being higher, the key generation, encryption and decryption time are being more. Similarly, the amount of energy consumption and most importantly, the amount of RAM consumption will increase. In the proposed method, we added two factors for checking write access and also expiring the users, which cause that the encryption and decryption time of proposed method will not be the same as Han's design. The results are represented in table 6.

7. Conclusion

In this paper, an encryption method based on a scalable and secure attribute is created to control the individual access to the information based on their access permission and also the possibility of expiring some users. In this method, with the new encrypted text that is intended to be sent to the user, the signature of the user who is allowed to write will also sent, and is performed by outsourcing a part of the encryption and decryption process to a fog server which is somehow reliable. Also, the two-part revocation methods are used to expire the individual access.
The factors of proposed method are as follows:
Revocation: Through an R list which is added to Access Policy, during the decryption and encryption of a small amount, the time of proposed method should be more. But the proposed method does not use oblivious transfer during the key generation, so the time of proposed method should be less to half or even more less, and of course because the Han method

uses the GBF method, it should have the more encryption and decryption time and its time be more than the proposed methods time.

Access time: Since it is checked that the user has the ability to write or read, we add it to the decryption phase, so it should increase the decryption time but it does not get to the Han design.

Reference

[1]. Li, Shancang, Li Da Xu, and Shanshan Zhao. "The internet of things: a survey." Information Systems Frontiers Vol.17, NO.2, PP.243-259, 2015.

[2]. Atzori, Luigi, Antonio Iera, and Giacomo Morabito. "The internet of things: A survey." Computer networks VOL.54, NO.15, PP. 2787-2805, 2010.

[3]. Oualha, Nouha, and Kim Thuat Nguyen. "Lightweight attribute-based encryption for the internet of things." In Computer Communication and Networks (ICCCN), 2016 25th International Conference on, PP. 1-6, 2016.

[4]. Peng, Chunyan, Xiujuan Du, Keqin Li, and Meiju Li. "An ultra-lightweight encryption scheme in underwater acoustic networks." Journal of Sensors, 2016.

[5]. Touati, Lyes, Yacine Challal, and Abdelmadjid Bouabdallah. "C-cp-abe: Cooperative ciphertext policy attribute-based encryption for the internet of things." In Advanced Networking Distributed Systems and Applications (INDS), International Conference on, PP. 64-69, 2014.

[6]. Touati, Lyes, and Yacine Challal. "Efficient cp-abe attribute/key management for iot applications." In Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing (CIT/IUCC/DASC/PICOM), IEEE International Conference on, PP. 343-350, 2015.

[7]. Zickau, Sebastian, Felix Beierle, and Iwailo Denisow. "Securing mobile cloud data with personalized attribute-based meta information." In Mobile Cloud Computing, Services, and Engineering (MobileCloud), 2015 3rd IEEE International Conference on, PP. 205-210, 2015.

[8]. Hur, Junbeom. "Improving Security and Efficiency in Attribute-Based Data Sharing." IEEE Trans. Knowl. Data Eng. Vol.25, NO. 10, PP. 2271-2282, 2013.

[9]. Jiang, Yinhao, Willy Susilo, Yi Mu, and Fuchun Guo. "Ciphertext-policy attribute-based encryption with key-delegation abuse resistance." In Australasian Conference on Information Security and Privacy, PP. 477-494, 2016.

[10]. Zhang, Yinghui, Dong Zheng, Xiaofeng Chen, Jin Li, and Hui Li. "Efficient attribute-based data sharing in mobile clouds." Pervasive and Mobile Computing Vol.28, PP.135-149, 2016.

[11]. Zhang, Yinghui, Xiaofeng Chen, Jin Li, Duncan S. Wong, Hui Li, and Ilsun You. "Ensuring attribute privacy protection and fast decryption for outsourced data security in mobile cloud computing." Information Sciences Vol.379, PP.42-61, 2017.

[12]. Han, Qi, Yinghui Zhang, and Hui Li. "Efficient and robust attribute-based encryption supporting access policy hiding in Internet of Things." Future Generation Computer Systems Vol.83, PP.269-277, 2018.

[13]. Borgh, Joakim, Edith Ngai, Börje Ohlman, and Adeel Mohammad Malik. "Employing attribute-based encryption in systems with resource constrained devices in an information-centric networking context." In Global Internet of Things Summit (GIoTS), PP. 1-6, 2017.

[14]. Singh, Meena, M. A. Rajan, V. L. Shivraj, and P. Balamuralidhar. "mq" In Communication systems and network technologies (CSNT), 2015 fifth international conference on, PP. 746-751, 2015.

[15]. Shi, Yanfeng, Qingji Zheng, Jiqiang Liu, and Zhen Han. "Directly revocable key-policy attribute-based encryption with verifiable ciphertext delegation." Information Sciences Vol.295, PP. 221-231, 2015.

[16]. Gupta, Harshit, Amir Vahid Dastjerdi, Soumya K. Ghosh, and Rajkumar Buyya. "iFogSim: A toolkit for modeling and simulation of resource management techniques in the Internet of Things, Edge and Fog computing environments." Software: Practice and Experience Vol.47, No. 9, PP. 1275-1296, 2017.