Ecc-bAsed ScalablE Revocation(EASER) scheme for selective sharing in resource-constrained devices

Divyashikha Sethia
Delhi Technological University Department of Computer Engineering
Raj Sahu (✉ rjsu26@gmail.com)
Delhi Technological University https://orcid.org/0000-0001-5160-0362
Sandeep Yadav
Delhi Technological University

Research Article

Keywords: Elliptic Curve Cryptography, IoT, CP-ABE, Scalable Revocation

Posted Date: July 1st, 2021

DOI: https://doi.org/10.21203/rs.3.rs-644632/v1

Noname manuscript No.
(will be inserted by the editor)

Ecc-bAsed ScalablE Revocation(EASER) scheme for selective sharing in resource-constrained devices

Divyashikha Sethia · Raj Sahu · Sandeep Yadav

Received: date / Accepted: date

Abstract IoT is one of the most promising technologies in modern industry. IoT devices are resource-constrained and hence require efficient and lightweight encryption schemes to provide data security. Fine-grained selective access control is required to share sensitive data with different stakeholders. There must be support for scalable revocation to revoke unsolicited users and provide uninterrupted access to valid users. In this paper, we propose a novel lightweight **E**cc b**A**sed **S**calabl**E** **R**evocation (EASER) CP-ABE scheme. It extends an existing CP-ABE-CSSK to mitigate a key collusion attack and extends it for scalable revocation without any need for any redistribution of keys, re-encryption of ciphertext, or requirement of any revocation lists beforehand. EASER CP-ABE scheme is lightweight and pairing-free with constant -size secret keys. Detailed qualitative and quantitative analysis of the EASER CP-ABE scheme proves that it outperforms the existing related schemes with acceptable storage and computational overheads.

Department of Computer Engineering, Delhi Technological University, New Delhi, India
· Divyashikha Sethia
E-mail: divyashikha@dtu.ac.in

Raj Sahu
E-mail: rjsu26@gmail.com

Sandeep Yadav
E-mail: samratrao101@gmail.com

Keywords Elliptic Curve Cryptography · IoT · CP-ABE · Scalable Revocation

1 Introduction

IoT has gained widespread utility in the healthcare sector. Resource-constrained implantable medical devices can be engrafted within a person for continuous monitoring and logging various parameters inside the body (Jovanov et al, 2005; Ullah et al, 2012). There can also be battery-powered portable health sensors gathering vitals for the patient and mobile devices to retain health information on a health card. These IoT devices can retain the medical vitals of a patient and can share with the desired healthcare professionals for accurate diagnostics and rehabilitation (Tang et al, 2006) as discussed in subsection 1.1. Since this sector is relatively new and evolving, many challenges are to be solved before a smooth, reliable system exists. One such challenge is to ensure privacy and security in a data-sharing network. Health-based IoT devices must encrypt all user's health data with limited access to any legal user. Standard encryption techniques such as symmetric and asymmetric algorithms can secure the data. However, both suffer from the drawback of sharing a common key with a group of users and lack selective access control. These encryption techniques are unsuitable for an environment where several users must access the encrypted data selectively based on their roles.

Attribute-Based Encryption (ABE) (Sahai and Waters, 2005) is a technique that efficiently supports one-to-many communication using public-key encryption. It defines a user's identity using a set of strings as descriptive attributes. ABE provides fine-grained access control of data by encrypting it with a set of attributes through which a data owner can specify the intended receivers. ABE schemes are classified into two variants : Key Policy Attribute-Based Encryption (KP-ABE) (Goyal et al, 2006; Sahai A, 2005; Attrapadung et al, 2011) and Ciphertext Policy Attribute-Based Encryption (CP-ABE) (Bethencourt et al, 2007). In a KP-ABE scheme, the user secret key associates with the access policy, while the ciphertext contains the attributes. The ciphertext can be decrypted by a secret key only if its access policy satisfies the ciphertext attributes. In a CP-ABE scheme, the user secret key contains the attributes while the ciphertext holds the access policy. A user can decrypt the ciphertext only if its secret key attributes satisfy the ciphertext's access policies. Since in CP-ABE schemes, the data owner has full control over access control, unlike the KP-ABE schemes where it has to trust the secret key distributor for accuracy of secret key policies, they are more suitable to provide fine-grained access control.

Many ABE schemes (Odelu et al, 2017b; Li et al, 2016, 2017) use bilinear pairings, which are computationally expensive. However, some ABE schemes are pairing free and use RSA (Odelu et al, 2017a), or Elliptic Curve Cryptography (ECC) (Odelu and Das, 2016). The key storage overhead of ECC-based schemes is around one-tenth of the RSA-based schemes. An ECC point multiplication is much more efficient than modular exponentiation or a bilinear mapping operation. Hence, due to an indispensable requirement of low storage and computational overheads, we consider the ECC-based ABE schemes in this paper.

Qin et al (2020) proposed a constant-time decryption scheme for Vehicular Ad-hoc Networks by outsourcing the decryption computation to a third-party server. However, the scheme is based on KP-ABE, and it has complex system complexity, which is dependent on a third-party server for the majority of storage and calculations. Ding et al (2018) proposed an ECC-based Pairing Free (PF-CP-ABE) scheme, which uses an attribute authority server to generate keys and revoke a given user or a particular attribute. However, this method brings too much dependency upon the proxy server, which should, ideally, assist the decryption only partially. Sowjanya and Dasgupta (2020) proposed a CP-ABE scheme based on Elliptic Curve Cryptography without bilinear pairing for Wireless Body Area Networks denoted as the WBAN-CP-ABE scheme in this paper. The WBAN-CP-ABE scheme (Sowjanya and Dasgupta, 2020) provides a similar feature of user revocation using a third-party server, putting many dependencies on the server. Odelu and Das (2016) proposed a CP-ABE-based Constant Sized Secret Key (CP-ABE-CSSK) scheme in an ECC setting with linear time complexity for both encryption and decryption. It provides constant storage for secret keys and is pairing-free. However, it is susceptible to key collusion attack (Herranz, 2017) due to the lack of entropy in the user secret keys.

Table 4 provides a detailed comparison between different ECC-based CP-ABE schemes for features and computational overheads, respectively. Although all the schemes offer linear time complexity for both encryption and decryption, the WBAN-CP-ABE scheme (Sowjanya and Dasgupta, 2020) has the least encryption time while the CP-ABE-CSSK scheme (Odelu and Das, 2016) has the least decryption time. For portable IoT devices where the read-to-write ratio is very high for a one-to-many broadcast system, we need efficient decryption with the least storage required for secret keys, while a slightly higher encryption overhead is tolerable. Hence, we consider extending the CP-ABE-CSSK scheme (Odelu and Das, 2016) in this paper.

Also, a secure resource-constrained system needs to have an efficient mechanism to revoke unsolicited users from the system without disrupting legitimate users' functioning (Sethia et al, 2017). Conventional methods require complete re-encryption and redistribution of keys, which becomes a bottleneck for the whole system, and interrupts access to valid users. None of the previous ECC-based CP-ABE schemes support scalable revocation(Sethia et al, 2017) which is essential for uninterrupted access to resource-constrained IoT devices.

1.1 Architecture

To further elaborate the requirement for selective sharing of health data and scalable revocation, we consider a healthcare system where different stakeholders get selective access to health information using resource-constrained medical devices, such as wearable health sensors and

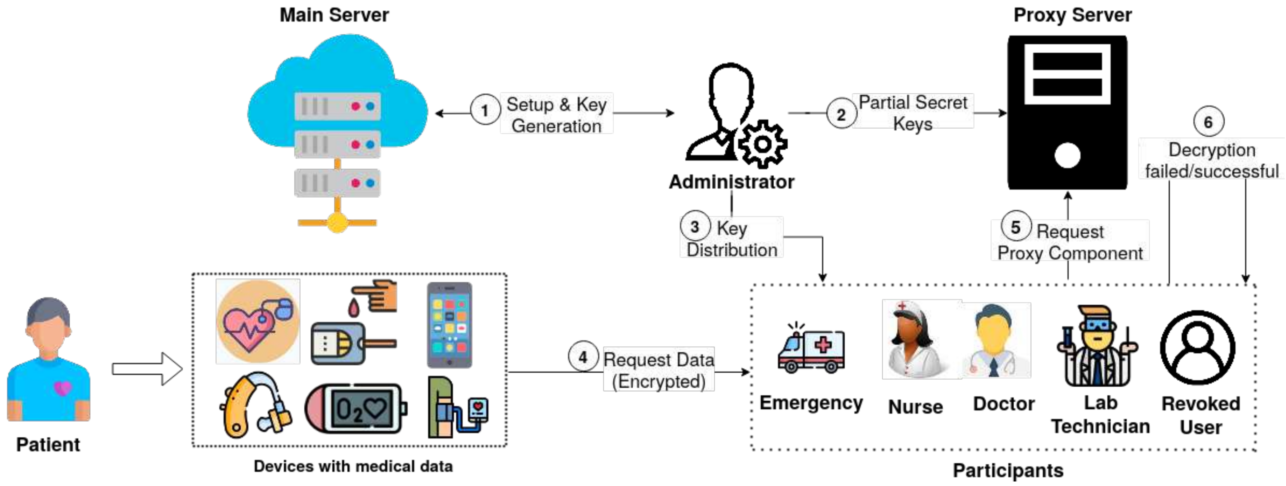


Fig. 1: Architecture for selective access from resource-constrained health devices

healthcard on the patient mobile device. Fig. 1 represents the architecture of the proposed healthcare system.

In this architecture, the system administrator assigns roles to each stakeholder and distributes their respective secret keys. It encrypts and stores the patient's health-related data in the patient's smartphone or the associated health sensors. A stakeholder can access the health information from the patient mobile device using low-energy wireless communication such as Bluetooth or Near Field Communication (NFC). After securing access to the health device's encrypted data, the stakeholder requests the Proxy server for proxy components required for partial decryption and scalable revocation. The proxy server retains the revocation list and sends the proxy component according to the revocation status of the user. The proxy component, when invalid, denies access to revoked users through the failure of the decryption process. It secures devices from adversaries and also allows uninterrupted access to valid users.

1.2 Our Contribution

The primary contributions made in this paper are as follows :

- Design and implementation of a novel lightweight proxy-based CP-ABE scheme called the **E**cc b**A**sed **S**calabl**E** **R**evocation (EASER) CP-ABE scheme for resource-constrained IoT and smart-card devices. It uses a proxy server to support partial decryption for scalable revocation (Sethia et al, 2017). It can revoke malicious users without interrupting valid users' access to medical devices.
- Mitigate an existing key collusion attack (Herranz, 2017) on the CP-ABE-CSSK (Odelu and Das, 2016) scheme.
- Detailed security analysis for the proposed EASER CP-ABE scheme.
- Use cases where the EASER CP-ABE scheme can best utilise the scalable revocation feature with selective access control for portable resource-constrained IoT devices.
- Detailed performance analysis of the EASER CP-ABE scheme to illustrate its lightweight feature.

1.3 Organization of paper

Section 2 presents the construction of the proposed EASER CP-ABE scheme. Section 3 gives a use case demonstrating a possible scenario where our scheme can prove to be effective. Section 4 gives a brief explanation of the existing key-collusion attack (Herranz, 2017), which we have mitigated in our EASER CP-ABE scheme. Section 5 discusses the analysis of our scheme against Chosen Ciphertext Attack (CCA), key-collusion attack, and replay attacks. Section 6 and section 7 concludes the paper with a quantitative analysis and conclusion respectively.

Table 1: List of notations

Symbol	Meaning
α, k_1, k_2	Private keys of system
p	Sufficiently large prime number
$\mathrm{E_p}$(a,b)	An elliptic curve $y^2 = x^3 + ax + b$ in $\mathrm{Z_p}$.
P	A base point in $\mathrm{E_p}$(a,b)
xP	ECC scalar multiplication
P+Q	Point addition on elliptic curve
$\mathrm{H_1}$,$\mathrm{H_2}$,$\mathrm{H_3}$,$\mathrm{H_4}$	Four collision-resistant hash functions
KDF	Key Derivation Function
$\mathbb{U}$	Universe of attributes
$\mathbb{A}$	Attributes set $\mathbb{A} \subseteq \mathbb{U}$
$\mathbb{P}$	Access policy $\mathbb{P} \subseteq \mathbb{U}$
$\mathbb{G}$	Elliptic curve group $\{p$,$\mathrm{E_p}$(a,b),P$\}$

Algorithm 1 Proxy Component access algorithm

```
Input: Connection request from a user device to a resource-constrained portable IoT device.
1. User device requests ciphertext from portable device.
2. The user device and portable device generates a
   new session key $K_{PS}$.
3. Portable device generates challenge $\mathbf{Ch} = E(k_{PS}, N_S)$,
   where $N_S$ : NONCE, $E()$ : an encryption function,
   and sends it to the user device.
4. The user device forwards the Challenge $\mathbf{Ch}$ and a request
   for the proxy component to the proxy server, which
   sends back the response $\mathbf{R} = E(K_{PS}, N_S - 1)$ and the
   respective proxy component.
5. User device keeps the proxy component and forwards
   $\mathbf{R}$ to portable device.
6. Upon successful validation of the response $\mathbf{R}$,
   the portable device sends the ciphertext to the user device.
7. User device proceeds with decryption of obtained ciphertext
   with the proxy component received in step 4.
8. The decryption completes if proxy component is correct,
   else aborts gracefully.
```

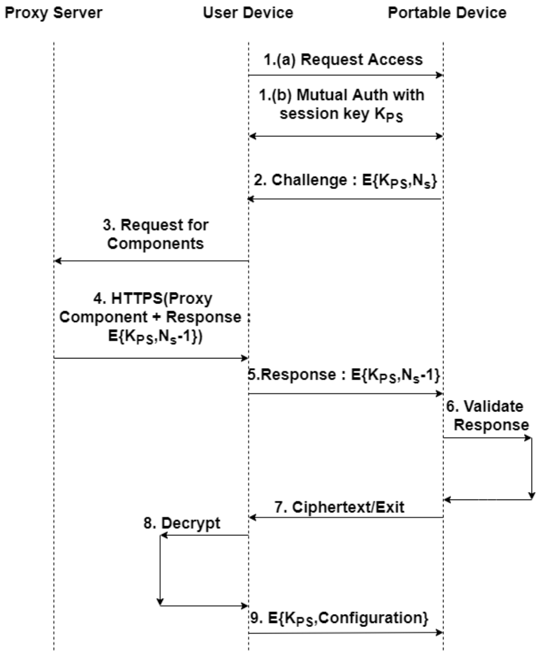

Proxy Server
User Device
Portable Device
1.(a) Request Access
1.(b) Mutual Auth with session key K_{PS}
2. Challenge : E{K_{PS},N_s}
3. Request for Components
4. HTTPS(Proxy Component + Response E{K_{PS},N_s-1})
5.Response : E{K_{PS},N_s-1}
6. Validate Response
7. Ciphertext/Exit
8. Decrypt
9. E{K_{PS},Configuration}

Fig. 2: Flow of events for partial decryption by Proxy Server

2 Proposed EASER CP-ABE Scheme

This work proposes a novel lightweigth Ecc-bAsed ScalablE Revocation (EASER CP-ABE) scheme. It improves Odelu and Das (2016)'s CP-ABE-CSSK scheme for a key collusion attack (Herranz, 2017) and also extends it to provide Scalable Revocation (Sethia et al, 2017). It uses a trusted Main Server and a Proxy Server.

Table 1 enumerates the notations for the proposed EASER CP-ABE scheme.

2.1 Proxy Server

The proxy server contains partial secret keys of all users interacting with the system. It also maintains a list of all revoked users. Whenever a user tries to decrypt, it must obtain a proxy component from the proxy server to complete the decryption process. The proxy components cause the failure of decryption for only revoked users. The proxy server delivers the proxy component to a user via a secured channel which is encrypted using a mutually generated session key. The proposed EASER CP-ABE scheme prevents replay attacks by using challenge-response between the proxy server and a user. Algorithm 1 shows how a user device seeks ciphertext from a portable device, along with proxy component from the proxy server, to access data. Fig. 2 further illustrates the flow of events for decryption with the help of the Proxy server.

2.2 Phases of the EASER CP-ABE scheme

2.2.1 Setup Phase

The setup phase is the initialisation phase. The size n of the universe of attributes $\mathbb{U}$ identifies how many different attributes are present in the secret keys and the ciphertext. A bit 1 or 0 denotes whether an attribute is present or absent respectively in the key; $n = 4$ signifies four attributes in a user's key and system policy that encrypts the data. A sample attribute definition can be {*'IT Professional?', 'joined before march 2016?', 'In an ongoing project?', 'Access to top-secret files?'*}. A sample attribute set can be $\mathbb{A} = \{1\ 0\ 1\ 0\}$. In the same way a sample policy for encryption can be $\mathbb{P} = \{0\,0\,1\,0\}$. In this case, since the attributes satisfy the given access policy, the user with this attribute set can successfully decrypt the ciphertext. If the policy is $\mathbb{P} = \{0\ 1\ 1\ 0\}$, then the user will not be able to decrypt since the attribute set $\mathbb{A}$ does not satisfy the Policy $\mathbb{P}$. Steps :

1. Choose the parameters for an Elliptic Curve group $\mathbb{G} = \{p, E_{\mathrm{p}}(a,b), P\}$, where P is the base point on the curve and p is some large prime number which defines the field Z_{p} of the curve. The order of the curve generated must be prime in order to allow ECC scalar division.
2. Within the finite field of p generate three random numbers α, k_1 and k_2 such that $\{\alpha, k_1, k_2\} \neq 0$. Calculate the following $\forall i \in 0, 1 \ldots n$:
$$P_{\mathrm{i}} = \alpha^{\mathrm{i}} P \tag{1}$$
$$U_{\mathrm{i}} = k_1 \alpha^{\mathrm{i}} P \tag{2}$$
$$V_{\mathrm{i}} = k_2 \alpha^{\mathrm{i}} P \tag{3}$$
3. Choose any four collision-resistant trapdoor hash functions H_1, H_2, H_3 and H_4 defined as :
$$H_1, H_4 \quad : \{0,1\}^* \longrightarrow {Z_{\mathrm{p}}}^* \tag{4}$$
$$H_2 \quad : \{0,1\}^* \longrightarrow \{0,1\}^{\sigma} \tag{5}$$
$$H_3 \quad : \{0,1\}^* \longrightarrow \{0,1\}^{|\mathrm{M}|} \tag{6}$$
where σ is some large random number, M is the message to be encrypted, $\mid x \mid$ denotes the length of a string x.
4. Using the above values generate the Global Secret Key (GSK) and the Global Public Key (GPK) $\forall i \in 0, 1, \ldots, n$ as follows:
$$GSK = \{\alpha, k_1, k_2\} \tag{7}$$
$$GPK = \{\mathbb{G}, P_{\mathrm{i}}, V_{\mathrm{i}}, U_{\mathrm{i}}, H_1, H_2, H_3, H_4\} \tag{8}$$

2.2.2 KeyGen Phase

This phase assigns user id (uid_j) to the user j. It takes as input the GSK, the GPK and the credentials of a user to generate a user secret key k_{p} corresponding to each uid_j.

1. Let $\mathbb{A} = \{a_1, a_2, \ldots, a_{\mathrm{n}}\}$ be the attribute set for the user. Compute :
$$f(\alpha, \mathbb{A}) = \prod_{i=1}^{n} (\alpha + H_4(i))^{(1-a_i)} \tag{9}$$
2. Within the finite field Z_{p}, generate two random numbers r_{u} and t_{u}. Compute the following values:
$$u_1 = r_u + k_1.f(\alpha, \mathbb{A}) \ \ (mod\ \ p) \tag{10}$$
$$u_2 = t_u - k_2.f(\alpha, \mathbb{A}) \ \ (mod\ \ p) \tag{11}$$
$$u_3 = r_u k_2 + t_u k_1 \ \ (mod\ \ p) \tag{12}$$
3. Send the user id uid_j and third user secret key as the component $X_i = u_3 \cdot f(\alpha, \mathbb{A})$ to the proxy server. The user's secret key k_u will be $k_u = \{u_1, u_2\}$.
4. Output the complete key k_p as $k_p = \{uid_j, k_u, \mathbb{A}\}$.

Algorithm 2 Encryption algorithm

```
Input: Plaintext, Access Policy, GPK
 1. Generate a random AES key K to encrypt the input
    plaintext, and the EASER CP-ABE encrypts the AES key.
2. Generate a random number $\sigma_m$.
3. Compute $\mathrm{r_m}$, $\mathrm{k_m}$ and $\mathrm{f(x,\mathbb{P})}$.
4. Compute $\mathrm{P_{m,i}}$, $\mathrm{K_{1,m}}$, $\mathrm{K_{2,m}}$, $C_{\sigma_m}$ and $C_m$.
6. Output the ciphertext C in a new file.
```

2.2.3 Encryption Phase

It takes the plaintext, access policy, and GPK as input. Standard AES algorithm encrypts the plaintext, and the EASER CP-ABE scheme encrypts the AES key.

1. Generate a random AES key K and use it to encrypt the input plaintext to produce an $AESciphertext.$
2. Generate a random number σ and another random number $\sigma_{\mathrm{m}} \in \{0{,}1\}^{|\,\sigma\,|}$.
3. Compute the following:
$$r_m = H_1(\mathbb{P}, K, \sigma_m) \tag{13}$$
$$k_m = KDF(r_m \mathbb{P}) \tag{14}$$
4. For the access policy $\mathbb{P} = \{b_1, b_2, \ldots, b_{\mathrm{n}}\}$, compute the following:
$$f(x, \mathbb{P}) = \prod_{i=1}^{n} (x + H_4(i))^{(1-b_i)} \tag{15}$$
The polynomial function $f(x, \mathbb{P})$ is of degree at-most n. Let f_{i} denote the coefficient of x_{i} in $f(x, \mathbb{P})$.
5. Compute the following:

$$P_{m,i} = r_m P_i, i = 1, \ldots, n - |\mathbb{P}| \tag{16}$$

$$\begin{aligned} K_{1,m} &= r_m \sum_{i=0}^{n} f_i U_i \\ &= r_m(f_0 U_0 + f_1 U_1 + \ldots + f_n U_n) \\ &= r_m(f_0 k_1 P + f_1 k_1 \alpha P + \ldots + f_n k_1 \alpha^n P) \\ &= r_m k_1 (f_0 + \alpha f_1 + \alpha^2 f_2 + \ldots + \alpha^n f_n) P \\ &= r_m k_1 f(\alpha, \mathbb{P}) P \end{aligned} \tag{17}$$

$$\begin{aligned} K_{2,m} &= r_m \sum_{i=0}^{n} f_i V_i \\ &= r_m(f_0 V_0 + f_1 V_1 + \ldots + f_n V_n) \\ &= r_m(f_0 k_2 P + f_2 k_2 \alpha P + \ldots + f_n k_2 \alpha^n P) \\ &= r_m k_2 (f_0 + \alpha f_2 + \alpha^2 f_2 + \ldots + \alpha^n f_n) P \\ &= r_m k_2 f(\alpha, \mathbb{P}) P \end{aligned} \tag{18}$$

$$C_{\sigma_m} = H_2(k_m) \oplus \sigma_m \tag{19}$$

$$C_m = H_3(\sigma_m) \oplus M \tag{20}$$

6. In a new file, output the ciphertext as C, where $C = \{AESciphertext, \mathbb{P}, P_{\mathrm{m,i}}, K_{1,\mathrm{m}}, K_{2,\mathrm{m}}, C_{\sigma_m}, C_{\mathrm{m}}\}$

Algorithm 2 summarizes the steps required in the encryption phase.

2.2.4 Decryption Phase

Algorithm 3 Decryption algorithm

```
Input: User secret key, ciphertext C
1. If P ⊈ A, abort
2. Compute U and V.
3. Send U, V along with uid from the user key to the proxy
   server.
4. Compute f(x,$\mathbb{P}$ )
5 Computer R = U+V.
        5.1 Q=$\frac{R}{X_{\mathrm{i}}}$, No revocation
        5.2 Q=$\frac{R}{B}$, Revocation.
6. Evaluate the expression for $F(x)$
7. Calculate the value of $W$.
8. Compute $T = \frac{1}{F_0}(Q - W)$.
        7.1 $T = r_{\mathrm{m}} P$, valid proxy component.
        7.2 $T \neq r_{\mathrm{m}} P$, tampered proxy component.
9. Compute $\sigma^{'}_{m}, M^{'}$ and $r^{'}_{m}$.
        8.1 If $r^{'}_{m} = r_{\mathrm{m}} P, M^{'}$ is original $M$
        8.2 If $r^{'}_{m} \neq r_{\mathrm{m}} P$, abort.
```

The decryption phase modifies the Odelu and Das (2016)'s CP-ABE-CSSK scheme's decryption phase for scalable revocation. It takes as input the ciphertext and a user's secret key to generate the AES key K. The AES key then decrypts the ciphertext and generates the original plaintext message.

1. If access policy $\mathbb{P}$ is not a subset of the attribute set $\mathbb{A}$, then abort.
2. Compute the values U and V as follows:

$$\begin{aligned} U &= u_2 K_{1,m} \\ &= (t_u - k_2 f(\alpha, \mathbb{A}))(r_m k_1 f(\alpha, \mathbb{P}) P) \end{aligned} \tag{21}$$

$$\begin{aligned} V &= u_1 K_{2,m} \\ &= (r_u + k_1 f(\alpha, \mathbb{A}))(r_m k_2 f(\alpha, \mathbb{P}) P) \end{aligned} \tag{22}$$

3. Send U, V along with uid to the proxy server. The proxy server sends a proxy component Q back to the user to assist partial decryption and scalable revocation.

Proxy Server:

It generates a proxy component and sends them to the client device to grant necessary access permissions. The proxy server maintains a list of user ids as well as a portion of their secret key components $(u_3 \cdot f(\alpha, \mathbb{A}))$ as discussed in the KeyGen phase 2.2.2. The proxy server does the following to generate the proxy component:

3.1. Compute

$$\begin{aligned} R &= U + V \\ &= (t_u - k_2 f(\alpha, \mathbb{A}))(r_m k_1 f(\alpha, \mathbb{P}) P) \\ &\quad + (r_u + k_1 f(\alpha, \mathbb{A}))(r_m k_2 f(\alpha, \mathbb{P}) P) \\ &= (t_u r_m k_1 f(\alpha, \mathbb{P})) P \\ &\quad - (r_m k_1 k_2 f(\alpha, \mathbb{P}) f(\alpha, \mathbb{A})) P \\ &\quad + (r_u r_m k_2 f(\alpha, \mathbb{P})) P \\ &\quad - (r_m k_1 k_2 f(\alpha, \mathbb{P}) f(\alpha, \mathbb{A})) P \\ &= r_m (r_u k_2 + t_u k_1) f(\alpha, \mathbb{P}) P \\ &= r_m u_3 F(\alpha) P \qquad [\textit{using equation } 12] \end{aligned} \tag{23}$$

3.2. The proxy server checks if the user id uid_j is registered and has not been revoked.

- **Case No Revocation** For a valid user, calculate proxy component Q as:

$$\begin{aligned} Q &= \frac{R}{X_i} \\ &= \frac{U + V}{u_3 f(\alpha, \mathbb{A})} = \frac{r_m u_3 f(\alpha, \mathbb{P}) P}{u_3 f(\alpha, \mathbb{A})} \\ &= r_m F(\alpha) P \end{aligned} \tag{24}$$

where $X_i = u_3 \cdot f(\alpha, \mathbb{A})$ is the partial key of the user, which is stored on the proxy server and $F(\alpha) = \frac{f(\alpha,\mathbb{P})}{f(\alpha,\mathbb{A})}$. The proxy server uses the users's stored key to compute and return the proxy component Q to the client.

- **Case Revocation** If user id uid_j is present in the revocation list, compute proxy component Q as :

$$Q = \frac{R}{B} \tag{25}$$

where B is some random number $\neq u_3 f(\alpha, \mathbb{A})$. Return Q to the user so that decryption fails, as shown in point 6.

4. Evaluate the expression for $F(x)$ as:

$$\begin{aligned} F(x) &= F(x, \mathbb{A}, \mathbb{P}) \\ &= \sum_{i=1}^{n-|\mathbb{P}|} (x + H_4(i))^{(a_i - b_i)} \end{aligned} \tag{26}$$

Let F_i be the coefficient of x_i in $F(x)$. Since $\mathbb{P} \subseteq \mathbb{A}$, $F_0 \geq 1$.

5. Calculate the value of W using the expression:

$$\begin{aligned} W &= \sum_{i=1}^{n-|\mathbb{P}|} F_i P_{m,i} \qquad (27) \\ &= r_m \left(\sum_{i=1}^{n-|\mathbb{P}|} F_i \alpha^{\mathrm{i}} \right) P \\ &= r_m \left(\sum_{i=1}^{n-|\mathbb{P}|} F_i \alpha^{\mathrm{i}} + F_0 - F_0 \right) P \\ &= r_m((F(\alpha) - F_0)P \\ &= r_m F(\alpha)P - r_m F_0 P \end{aligned}$$

6. Compute $T = \frac{1}{F_0}(Q - W)$.

For valid proxy component

$$\begin{aligned} T &= \frac{1}{F_0}(Q - W) \\ &= \frac{1}{F_0}(r_m F(\alpha)P - (r_m F(\alpha)P - r_m F_0 P)) \\ &= \frac{1}{F_0}(r_m F_0 P) \end{aligned}$$

$$\therefore T = r_m P \tag{28}$$

Invalid proxy component for revocation

$$\begin{aligned} T &= \frac{1}{F_0}(Q - W) \\ &= \frac{1}{F_0}\left(\frac{R}{B} - W\right) \\ &= \frac{1}{F_0}\left(\frac{(r_m u_3 f(\alpha, \mathbb{A})P)}{B} - W\right) \end{aligned}$$

$\because$ B is some random number

$$\therefore T \neq r_m P \tag{29}$$

7. Compute:

$$\sigma_m^{'} = H_2(KDF(T)) \oplus C_{\sigma_m} \tag{30}$$

$$K^{'} = C_m \oplus H_3(\sigma_m^{'}) \tag{31}$$

$$r_m^{'} = H_1(\mathbb{P}, M^{'}, \sigma_m^{'}) \tag{32}$$

If $r_m^{'} P = T$ (obtained in point 6) then message $K^{'}$ is the original K, else abort with failure status.

8. Treat the obtained $K^{'}$ as the AES-key to decrypt the ciphertext, given as input in the Decryption Phase, to produce the expected original file.

Algorithm 3 summarizes the steps required in the decryption phase.

3 Use case

The COVID-19 pandemic brought new instances of security threats within the premises of the smart healthcare sector. Today, a single complete medical profile data is worth more than an individual's social security details. Several security breaches have leaked medical data impacting numerous people. The EASER CP-ABE scheme (Fig. 1) can prove to be vital in securing medical devices and medical data in a resource-constrained environment. By employing selective data access, the attack surface of the whole system narrows down. Also, if a stakeholder is no more a valid user, the efficient user revocation functionality easily restricts that user from further accessing any data in the system.

3.1 Application of EASER CP-ABE scheme

A patient's smart-card-based portable health folder (Sethia et al, 2014) can contain data in the form of prescriptions, reports, medication lists from different hospitals as well as body sensors as shown in the architecture in Fig. 1 in section 1. A valid health professional can access the data as per the authorized roles.

Table 2: Role-based read/write access to data for various stakeholders

Stakeholder	Vitals	Diseases	Medicines	Lab
Nurse	RW	R	R	R
Doctor	RW	RW	RW	RW
Lab Technician	–	–	–	RW
Pharmacist	–	–	RW	–
Emergency	RW	RW	RW	RW

Table 3: Sample Write-policy for various stakeholders

Stakeholder	Vitals	Diseases	Medicines	Lab
Nurse	1	0	0	0
Doctor	1	1	1	1
Lab Tech	0	0	0	1
Pharmacist	0	0	1	0
Emergency	1	1	1	1

Consider various sample stakeholders : Nurse, Doctor, Lab Technician, Pharmacist, and Emergency; who can access the health data from medical devices directly. Table 2 describes a sample role-based access structure in which various stakeholders have varying read/write access to data from different sections of the health data. The EASER CP-ABE scheme encrypts the health data to differentiate between a read and a write request using the method proposed by Sethia et al (2017).

Say, a Patient (P) arrives at a hospital complaining of restlessness and chest pain. A nurse(N) first reads about P 's past diseases and her medicines, using her secret keys. After measuring P 's temperature and updating the vitals, the assigned doctor (D) immediately recommends a COVID-19 and other lab tests. A lab technician (L) performs the test and writes the data to P 's healthfolder using his secret keys. After confirming a case of COVID-19, D writes the findings into the Diseases section and introduces P to the first batch of medication using his write access for the Medicines section. A pre-configured robot (R) delivers the prescribed medicines on time to P and sanitise her room regularly. Note that R has no access to read lab reports or different diseases a patient was/is suffering. The same goes for a Pharmacist who has minimal access to any of the patient's health data. The whole process of reading/modifying private data can be made contactless using NFC-based modules or Bluetooth.

Table 3 provides a sample write-policy which, when combined with AND-based access-tree structure, will create a restrictive environment for data similar to the situation depicted above. If the policy $\{01001\}$ encrypts the Disease section, then only the doctor can decrypt it using its attribute set $\{1111\}$. Any stakeholder can be revoked and denied access to the medical information by the system administrator, who can request the proxy server to mark that person as revoked. As explained in section 2.2.4, the proxy server provides the wrong proxy component to a revoked user leading to failure in data decryption. Hence, the proposed EASER CP-ABE scheme allows scalable revocation and uninterrupted access to valid users by fulfilling all constraints for scalable revocation.

4 Attack on CP-ABE-CSSK (Odelu and Das, 2016) scheme

Herranz (2017) demonstrated how Odelu and Das (2016)'s CP-ABE-CSSK scheme is vulnerable to key collusion attack by a group of attackers. This attack permits the adversaries to collude existing secret keys to generate a new valid key.

In the CP-ABE-CSSK scheme, a user secret key has two components u_1 and u_2 as :

$$u_1 = r_u + k_1 t_u (mod\ p)$$
$$u_2 = s_u - k_2 t_u (mod\ p)$$

$k_u = (u_1, u_2)$ becomes the final user secret key. According to Herranz (2017), the secret key k_u lacks enough randomness i.e. despite r_u and t_u being random, the final values u_1 and u_2 are not independent as shown in equation 33.

$$u_2 = -\frac{k_2}{k_1} u_1 + \frac{1}{k_1 f(\alpha, \mathbb{A})} (mod\ p) \tag{33}$$

The adversary makes two queries in the form of equation 33 to compute $X = -\frac{k_2}{k_1}$ $(mod\ p)$ and $Y_{\mathbb{A}} = \frac{1}{k_1 f(\alpha,\mathbb{A})}$ $(mod\ p)$ for an attribute set $\mathbb{A}$.

The attack is divided into three steps:

1. **Obtain values of X and $Y_{\mathbb{A}}$**
   Make queries for attribute sets $\mathbb{A}_1$, $\mathbb{A}_2$ and $\mathbb{A}_3$ and compute sets (X,$Y_{\mathbb{A}_1}$), (X,$Y_{\mathbb{A}_2}$) and (X,$Y_{\mathbb{A}_3}$).

2. **Generate $Y_{\mathbb{B}}$ using $Y_{\mathbb{A}}$'s** As the function $f(x, \mathbb{A})$ is product of multiple terms, for different attribute sets, we can combine them to compute new $f()$ values. Say, if $n = 3$, $\mathbb{P} = \{100\}$ and $\mathbb{A}_1 = \{001\}$, $\mathbb{A}_2 = \{110\}$, $\mathbb{A}_3 = \{010\}$. For an attribute set $\mathbb{B} = \{101\}$, which satisfies the policy $\mathbb{P}$, the following equality holds :

$$f(\alpha, \mathbb{B}) = \frac{f(\alpha, \mathbb{A}_1).f(\alpha, \mathbb{A}_2)}{f(\alpha, \mathbb{A}_3)} \ mod\ p$$

   Since $\mathrm{Y}_{\mathbb{B}}$ is related to $f(\alpha, \mathbb{B})$, we get the value of $Y_{\mathbb{B}}$ in terms of $Y_{\mathbb{A}_1}, Y_{\mathbb{A}_2}$ and $Y_{\mathbb{A}_3}$.
3. **Produce a new key**
   (a) Use the value of $(X, Y_{\mathbb{B}})$, to generate the secret key k_u by choosing a random value of u_1 and generate u_2 using equation 33.
   (b) Use this secret key to perform successful decryption of the ciphertext.

   The proposed EASER CP-ABE scheme mitigates this attack as discussed in the following section 5.

5 Security Analysis

5.1 Selective Game

CP-ABE selective game proves the security strength of corresponding CP-ABE schemes. The Chosen Ciphertext Attack (CCA) test for indistinguishability of two messages encrypted using a CP-ABE scheme under a chosen-plaintext attack (IND-CPA). The game between an adversary $\mathscr{A}$ and a challenger $\mathscr{B}$ is similar as in the CP-ABE-CSSK scheme (Odelu and Das, 2016):

- **Initialisation:** $\mathscr{A}$ declares an n-bit challenge access policy $\mathbb{P}$' using which the encryption is to be performed.
- **Setup:** $\mathscr{B}$ runs the Setup phase to obtain the (GSK, GPK) pair. It then sends the GPK to $\mathscr{A}$.
- **Query:** $\mathscr{A}$ can make multiple queries for decryption keys k_{u^i} corresponding to attribute sets $\mathbb{A}_1, \mathbb{A}_2, \ldots, \mathbb{A}_\omega$ subjected to a condition that no attribute set should satisfy the access policy $\mathbb{P}$'.
- **Challenge:** $\mathscr{A}$ generates two plaintexts M_0 and M_1 having equal lengths and submits it for the challenge. $\mathscr{B}$ randomly chooses a bit $b \in \{0, 1\}$ by flipping a fair coin and replies by sending the ciphertext generated after encrypting message M_b with $\mathbb{P}$'.
- **Query:** The query is repeated multiple times with different attribute sets.
- **Guess:** $\mathscr{A}$ guesses b_g, the value of b in the challenge round, and wins the game if b_g= b.

In this game the advantage ϵ of $\mathscr{A}$ is defined by ϵ = P[b_g= b] - $\frac{1}{2}$

5.2 Security against threats

We analyse the proposed EASER CP-ABE scheme's security, which is an extension to the CP-ABE-CSSK (Odelu and Das, 2016) scheme. The primary goal is to establish that this scheme is resistant to key collusion attacks.

- **Security against key-collusion attack:** EASER CP-ABE scheme fixes the attack Herranz (Herranz, 2017) in the CP-ABE-CSSK (Odelu and Das, 2016) scheme. The secret key $k_u = (u_1, u_2)$ given to a user is calculated using the equations:

$$u_1 = r_u + k_1 \cdot f(\alpha, \mathbb{A}) \quad (34)$$

$$u_2 = t_u - k_2 \cdot f(\alpha, \mathbb{A}) \quad (35)$$

  where r_u and t_u are random values, k_1 and k_2 are system private keys common to all users and $f(\alpha, \mathbb{A})$ is a value constant for all users having the same attribute set $\mathbb{A}$. By re-arranging the equations, we get

$$u_2 = -\frac{k_2}{k_1}u_1 + \frac{k_2}{k_1}r_u + t_u \quad (36)$$

  Substituting X for $-\frac{k_2}{k_1}$ and $\frac{k_2}{k_1}r_u + t_u$ by γ we get

$$u_2 = u_1 X + \gamma \quad (37)$$

  In the equation 37, u_2 and u_1 is known to an adversary, X is constant but unknown while γ is variable for each user. For any system of l such equations, there will always be $l + 1$ unknowns. Hence, there is no deterministic way to solve for X.
  Also, the equations 34 & 35 form a system of l linear equations with $l + 2$ unknowns. Therefore, inorder to solve for k_1 and k_2, the attacker needs to correctly guess r_u, t_u and $f(\alpha, \mathbb{A})$ which, given the large range of input possible, is highly improbable. As an another line of defense, the EASER CP-ABE scheme escrows a part of the user secret keys u_3 to the proxy server so that even if a group of attackers succeed in generating a new valid secret key k_u, the corresponding u_3 will always be secure considering the absence of any method to determine all of r_u, t_u, k_1 and k_2 simultaneously to calculate u_3 as

Table 4: Comparison of theoretical time complexity

Scheme	Encryption	Decryption
WBAN-CP-ABE	$(\|\mathbb{P}\|+1)T_{ecm}\mathbb{G}$	$2\|\mathbb{P}\|T_{ecm}\mathbb{G}$
PF-CP-ABE	$(4n+1)T_{ecm}\mathbb{G}$	$(\|\mathbb{P}\|+1)T_{ecm}\mathbb{G}$
CP-ABE-CSSK	$(3n- \mid \mathbb{P} \mid +5)T_{ecm}\mathbb{G}$	$(n- \mid \mathbb{P} \mid +3)T_{ecm}\mathbb{G}$
Proposed (EASER CP-ABE)	$(3n- \mid \mathbb{P} \mid +5)T_{ecm}\mathbb{G}$	$(n- \mid \mathbb{P} \mid +6)T_{ecm}\mathbb{G}$

$\mid \mathbb{P} \mid$:Length of Policy; n :Number of attributes in the system; T_{ecm} :Time of one ECC scalar multiplication; $\mathbb{G}$:Elliptic Curve Group

$$u_3 = r_u k_2 + t_u k_1$$

Hence, the proposed EASER CP-ABE scheme is secure from Key Collusion attacks.

- **Resistance to replay attacks:** In the proposed EASER CP-ABE scheme, a secure HTTPS channel helps generate a session key between the User Device and the resource-constrained IoT portable Device. Both use the HTTPS session key to exchange challenge and response using a randomly generated NONCE. The portable device sends a challenge, including the nonce, to the proxy server via the user device. The proxy server then sends a response and the proxy components to the user device. The user device further keeps the proxy component and forwards the response to the portable device to validate it. The portable device validates the response to ensure there is no replay attack to access the ciphertext and then forwards the ciphertext to the user device. Hence, the User Device cannot replay the old proxy component to decrypt Portable Device's ciphertext.

6 Performance Analysis

This work implements the proposed EASER CP-ABE and the CPABE-CSSK (Odelu and Das, 2016) schemes on an Intel (R) Core (TM) i5-7200U CPU @ 2.50GHz quad-core processor. Table 4 compares the theoretical time complexity of Encryption and Decryption for different ECC-based CP-ABE schemes.

The estimation of the time complexity of the proposed EASER CP-ABE scheme's Encryption phase considers every step and counts the number of ECC scalar multiplication operations performed. In section 2.2.3, equation (17) contributes 1 scalar multiplication, equation (19) gives $(n - \mid \mathbb{P} \mid)$ multiplications, equation (20) and equation (21) each contribute $n + 2$ multiplications. This way, for the Encryption phase, we have a total $(3n - \mid \mathbb{P} \mid + 5)$ ECC scalar multiplications. Since the encryption phase of the EASER CP-ABE scheme is similar to that of the CP-ABE-CSSK scheme, both schemes have an encryption time complexity of $(3n - \mid \mathbb{P} \mid + 5)T_{ecm}\mathbb{G}$.

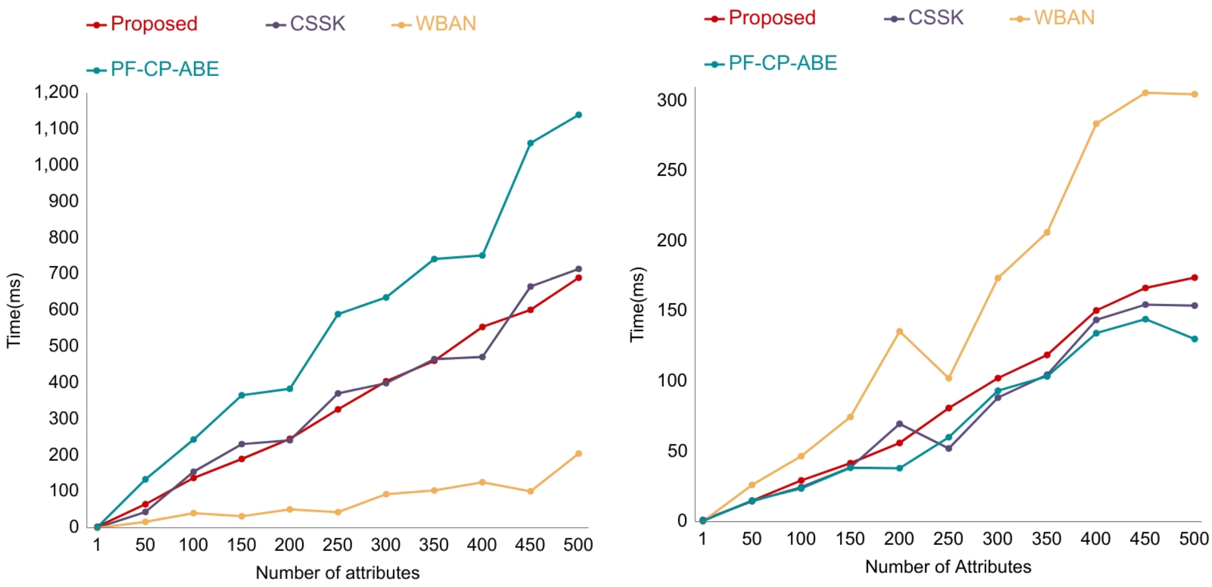
Table 5: Comparison of computation times

Scheme	Encryption	Decryption
WBAN-CP-ABE	$275.55ms$	$550ms$
PF-CP-ABE	$2200.55ms$	$275.55ms$
CP-ABE-CSSK	$1377.75ms$	$276.65ms$
Proposed (EASER CP-ABE)	$1377.75ms$	$278.30ms$

In the case of decryption, step 2 contributes two multiplications, and step 3.2 needs one multiplication (scalar division is effectively a modular inverse followed by scalar multiplication) for both revocation and no-revocation case; step 5 contributes $(n - \mid \mathbb{P} \mid)$, point 6 needs one, and step 7 needs additional two multiplications. This way, we have total $(n - \mid \mathbb{P} \mid + 6)$ ECC scalar multiplications in the decryption phase.

For performance comparison, let the number of attributes, $n = 1000$, the number of bits in $\mathbb{A} = 600$ and the number of bits in $\mathbb{P} = 500$. Table 5 uses these values and compares computation times for the different ECC-based CP-ABE schemes. The decryption time for EASER CP-ABE scheme is around the same or less than the other CP-ABE schemes and has the feature of Constant Sized Secret Keys and Scalable User Revocation.

Fig. 3 illustrates the impact of the number of attributes on the execution times of the Encryption and the Decryption phase of various discussed CP-ABE schemes. For each number of attributes, n, we take the length of policy, $\mid \mathbb{P} \mid$, as half of the corresponding n value i.e. $n/2$. Results indicate that the decryption time of the proposed EASER-CP-ABE scheme is very similar to CP-ABE-CSSK scheme (Odelu and Das, 2016) scheme. The WBAN-CP-ABE scheme (Sowjanya and Dasgupta, 2020) performs best in Encryption and worst in decryption. Since the work focuses on the feasibility of decryption on resource-constrained device that access the IoT device, lower decryption time is required. Also as compared to the other schemes, the dependency of EASER



(a)

(b)

Fig. 3: Comparison of execution time of (a) Encryption Phase and (b) Decryption Phase of the ECC-based CP-ABE schemes

Table 6: Comparison of Secret Key size for 80-bits of security

Scheme	Secret Key (bits)	Complexity
WBAN-CP-ABE	$80 \cdot n$	Linear
PF-CP-ABE	$160 \cdot n$	Linear
CP-ABE-CSSK	320	Constant
Proposed (EASER CP-ABE)	320	Constant

n: number of attributes in the system

CP-ABE Scheme on the third party server is minimal only for partial decryption. None of the other schemes except EASER-CP-ABE scheme support scalable revocation.

Table 6 compares the impact of the number of attributes on the storage for the user's secret keys. The WBAN-CP-ABE (Sowjanya and Dasgupta, 2020) and the PF-CP-ABE (Ding et al, 2018) schemes have key size linearly proportional to the number of attributes. Hence, for a moderately large value of n, the space requirement will become a bottleneck considering storage constraints on the portable IoT device. The proposed EASER CP-ABE scheme has constant-sized keys, which makes it well suited for practical portable resource-constrained IoT devices.

Hence, the proposed EASER CP-ABE scheme provides scalable revocation and attack mitigation. Furthermore, the minimal computational and storage overhead makes our proposed scheme very battery efficient for a resource-constrained device.

7 Conclusion and Future Scope

This paper presents an ECC-based lightweight EASER CP-ABE scheme for smart-card-based resource-constrained portable IoT devices to provide selective access-control and scalable revocation with the help of a proxy server. It provides uninterrupted access to valid users, even on the revocation of adversaries, without any overheads. Unlike previous ECC-based CP-ABE schemes, the EASER CP-ABE scheme has significantly less dependency on the server because it only stores a portion of the user's secret keys with minimal computations required to produce the proxy component required for each decryption. The EASER CP-ABE scheme uses ECC point multiplication instead of complex bilinear-pairings, because of which the battery requirements are also low. It has constant-sized secret keys, limiting the storage requirement for each user's secret keys irrespective of the number of attributes defined in the system. The security analysis proves that the proposed scheme is resistant to key collusion attacks and replay attacks. The EASER CP-ABE can benefit the smart-medicare industry, where smart body-sensors continuously interact with multiple users and store all the real-time data into a personal portable smart-card-based health folder. Various healthcare professionals can access this data based on the roles they are designated.

In the future, we intend to implement the proposed EASER CP-ABE scheme on Single Board Computer (SBC) based Raspberry Pi and smart cards to fine-tune it further for practical applications and flexible user interaction. In addition, we also plan to introduce a mechanism for conditional authentication(Qin et al, 2020) which can serve both the purpose of integrity and authenticity in the system.

Acknowledgements We thank Samsung India Electronics Private Limited for providing the Samsung Innovation Research center at Delhi Technological University (DTU), in which we pursued this research. We also thank Lokesh N. Shankar and Ram Kumar, who contributed to some of the proposed algorithm's software implementation.

Declarations

Funding Not Applicable

Conflict of Interest The authors declare that they have no conflict of interest.

Availability of data and material Not Applicable

Code availability The code that support the findings of this study are available from the corresponding author, upon reasonable request.

References

Attrapadung, et al (2011) Expressive Key-Policy Attribute-Based Encryption with constant-size ciphertexts. In 14th International Conference on Practice and Theory in Public Key Cryptography pp 90–108

Bethencourt J, Sahai A, Waters B (2007) Ciphertext-policy Attribute-Based Encryption. pp 321–334, DOI 10.1109/SP.2007.11

Ding S, Li C, Li H (2018) A novel efficient Pairing-Free CP-ABE based on Elliptic Curve Cryptography for IoT. IEEE Access 6:27336–27345

Goyal V, Pandey O, Sahai A, Waters B (2006) Attribute-Based Encryption for fine-grained access control of encrypted data. In: Proceedings of the 13th ACM conference on Computer and communications security, pp 89–98

Herranz J (2017) Attribute-Based Encryption implies identity-based encryption. IET Information Security 11(6):332–337, DOI 10.1049/iet-ifs.2016.0490

Jovanov, et al (2005) A Wireless Body Area Network of intelligent motion sensors for computer assisted physical rehabilitation. Journal of NeuroEngineering and Rehabilitation 2(1):1–10

Li J, Yao W, Han J, Zhang Y, Shen J (2017) User collusion avoidance cp-abe with efficient attribute revocation for cloud storage. IEEE Systems Journal 12(2):1767–1777, DOI 10.1109/JSYST.2017.2667679

Li L, Chen X, Jiang H, Li Z, Li KC (2016) P-cp-abe: parallelizing ciphertext-policy attribute-based encryption for clouds. In: 2016 17th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), IEEE, pp 575–580, DOI 10.1109/SNPD.2016.7515961

Odelu V, Das AK (2016) Design of a new CP-ABE with constant-size secret keys for lightweight devices using Elliptic Curve Cryptography. Security and Communication Networks 9(17):4048–4059

Odelu V, et al (2017a) Expressive CP-ABE scheme for mobile devices in IoT satisfying constant-size keys and ciphertexts. IEEE Access 5:3273–3283

Odelu V, et al (2017b) Pairing-based CP-ABE with constant-size ciphertexts and secret keys for cloud environment. Computer Standards & Interfaces 54:3–9

Qin, et al (2020) An ECC-based access control scheme with lightweight decryption and conditional authentication for data sharing in vehicular networks. Soft Computing 24(24):18881–18891

Sahai A, Waters B (2005) Fuzzy Identity-Based Encryption. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques, Springer, pp 457–473

Sahai A WB (2005) Applications of cryptographic techniques. In 25th Annual International Conference on the Theory and Applications of Cryptographic Techniques p 457–473

Sethia, et al (2014) NFC based secure mobile healthcare system. In: Communication Systems and Networks (COMSNETS), 2014 Sixth International Conference on, IEEE, pp 1–6

Sethia, et al (2017) CP-ABE for selective access with scalable revocation: A case study for mobile-based healthfolder. International Journal of Network Security, Vol20, No4 pp 689–701

Sowjanya K, Dasgupta M (2020) A Ciphertext-Policy Attribute-Based Encryption scheme for Wireless Body Area Networks based on ECC. Journal of Information Security and Applications 54:102559

Tang, et al (2006) Personal Health Records: definitions, benefits, and strategies for overcoming barriers to adoption. Journal of the American Medical Informatics Association 13(2):121–126

Ullah S, et al (2012) A comprehensive survey of Wireless Body Area Networks. Journal of medical systems 36(3):1065–1094