Journal of Cloud Computing:
Advances, Systems and Applications

RESEARCH

Open Access

An efficient and scalable vaccine passport verification system based on ciphertext policy attribute-based encryption and blockchain

Somchart Fugkeaw[1*]

Abstract

Implementing a trust and secure immunity or vaccine passport verification system is now crucial for many countries. The system typically aims to enable the secure access control and verification of vaccination records which will be used by trusted parties. However, the issues related to the system scalability in supporting a large number of data access requests, the enforcement of the user consent for data sharing, and the flexibility in delegating the access capability to trusted parties have not been resolved by existing works. In this paper, we propose a Universal Vaccine Passport Verification System (UniVAC) to support a decentralized, scalable, secure, and fine-grained, access control for Covid-19 vaccine passport data sharing and verification. At a core of our scheme, we employ the ciphertext policy attribute-based encryption (CP-ABE) to support secure and fine-grained access control and use the blockchain to record access transactions and provide data indexing. Furthermore, we propose a ciphertext retrieval method based on regional blockchain segmentation and introduce the outsourced CP-ABE decryption as a part of the proxy re-encryption (PRE) process to enable scalable and secure ciphertext delivery of the encrypted vaccine passport under the requestor's public key. Finally, we conducted the extensive experiments in real cloud environment and the results showed that our proposed scheme is more efficient and scalable than related works.

Keywords Vaccine passport, Access control, Blockchain, CP-ABE, Proxy Re-encryption, Smart contract

Introduction

The outbreak of Coronavirus-19 has caused millions of hospitalizations and deaths worldwide. Each country tries its best to reduce the number of deaths. One of the most efficient ways to curb the danger of the virus is by letting each country vaccinate its citizens with a trustable department. It is visible that after each nation provides the vaccine to its citizens, the number of deaths has been dropping.

Although vaccination has been intensively invoked to alleviate the outbreak, some people do not want to get vaccinated. This is because there are controversies in some countries regarding the effectiveness and long-term effect of the vaccines. To encourage their people to get vaccinated, many countries have exposed the strategies to enforce their citizens to get vaccinated through the issuance of certificates of vaccination. Providing the efficient verification of an individual's COVID-19 vaccination would improve cross-border mobility and limit the outbreak of the virus [1].

People who have a certificate depicting the qualified types of vaccine and sufficient quantity will gain several advantages in using public services. Specifically, the restrictions on the people to enter or leave the countries could be relaxed. Most countries issue the Covid-19 vaccine passport in either the form of hard copy or electronic format. In the latter technique, the certificate is usually encrypted and stored in the cloud and it can be simply

*Correspondence:
Somchart Fugkeaw
somchart@siit.tu.ac.th
[1] Sirindhorn International Institute of Technology, Thammasat University, Pathum Thani 12000, Thailand

retrieved via QR code or barcode. Even though the electronic vaccine passport seems usable as valid proof, the interoperability and the trust of the vaccine passport is still an issue. This is because it lacks the valid verification for guaranteeing the authenticity and integrity of the vaccine passport. To deal with this problem, an efficient and reliable vaccine verification scheme is essential.

Existing works [1–5] have generally employed cloud computing and blockchain to construct the secure vaccine passport sharing and verification. However, they have the following common issues:

(1) they do not fully support fine-grained and scalable access control with flexibility for granting third parties the access right to vaccine passports.
(2) they do not support the consent of the vaccine record's owner before the passport is published and shared.
(3) they do not provide the mechanism that enables trust between the international immigration offices and system entities.

Regarding the first problem, most approaches emphasize the privacy of the vaccine certificate where the personal data and vaccination records are encrypted and stored in the data outsourcing environment such as a cloud storage. They generally overlook the scalability and the performance of the system in supporting the large number of verification queries. The decryption delegation to other trusted parties has not been supported in a scalable manner. Second, existing access control and verification schemes have not taken the consent of the vaccination records' owners into their data sharing protocol. This brings the issue related to data privacy compliance mandated by data privacy regulations such as General Data Protection Regulation (GDPR). Therefore, most solutions cannot be implemented in real practice. Finally, the provable security and trust mechanism between the requestors such as immigration offices and system entities such as cloud, and blockchain have not been incorporated in their design.

To encounter the privacy issue of the vaccine passport information, many research works have proposed the access control systems that support secure data sharing with the provision of authentication and encryption. Importantly, the data sharing is usually done through the blockchain and cloud storage. Most works leverage blockchain to support user authentication and store access transactions, and use cloud storage to keep the encrypted form of identity documents and the digital health passport. Regarding the cryptographic methods used to support the data privacy, traditional encryption methods such as symmetric encryption and public key encryption are not suitable to support data sharing in such environment since the symmetric encryption renders expensive key management cost while the PKI yields the overheads in dealing with multiple copies of ciphertexts.

Consequently, most works opt to consider other cryptographic-based access controls that offer both encryption and access control enforcement. Ciphertext policy attribute-based encryption (CP-ABE) proposed by Bethencourt [6] has been considered as an effective solution for supporting a secure and fine-grained access control for data outsourcing environment. To date, CP-ABE has been adopted by many cloud-based access control schemes. In CP-ABE, the data owner encrypts the data by using the access policy constructed from the formulation of a set of attributes and logical threshold gates OR, AND, and MofN. Data users who have a secret key that satisfies the access policy structure can decrypt the ciphertext.

In a cloud-based and blockchain-based access control, a symmetric encryption and CP-ABE are usually employed for fulfilling the privacy-preserving access control solution [7]. Specifically, existing works applying blockchain in their access control schemes [8–15] have a focus on improving the performance of the authentication and the auditability. These benefits could be a trade-off with the incurred communication cost related to the authentication and verification process occurring in the blockchain, cloud, and applications.

To the best of our knowledge, there are no blockchain-based access control schemes that provide the flexibility of the decryption of data access to authorized parties based on the data owner's consent with fully traceable feature. Essentially, the delegation of decryption capability to authorized parties helps reduce the dependency on the availability of data owners as well as creates the trust of the data access verification. Such feature is even crucial for the data sharing service in which there is an authority that wants to verify or access the sensitive data through the trusted system rather than getting the confirmation from the data owner. Furthermore, there are no works that explicitly demonstrate the practicality in supporting the large volumes of verification requests. Accordingly, this paper provides the first attempt in delivering practical and secure Covid 19 vaccination data sharing and verification to the concern parties such as international immigration offices of many countries.

To this end, we proposed the design and the development of a universal vaccine passport verification system called UniVAC that aims to address all the above issues and entails an efficient and reliable access control and verification scheme. The UniVAC system is based on the integration of the CP-ABE, blockchain technology,

proxy re-encryption (PRE), and cloud computing to enable privacy-preserving data and efficient sharing and verification of vaccine records. In our scheme, we designed a ciphertext retrieval method based on regional segmentation and introduced fully outsourced CP-ABE decryption as a part of proxy re-encryption process. The concept of outsourced decryption offloads expensive CP-ABE computation costs related to pairing and exponentiation cost to be executed by the semi-trusted proxy while the secrecy of the ciphertext is not compromised. In essence, our proposed PRE technique transforms the ciphertext designated for the requestor in the different access control domain to support secure delegation through the delivery of the encrypted vaccine passport under the requestor's public key. In addition to the computation offload and secure transformation of the ciphertext into another form under the authorized recipient, combining PRE with CP-ABE and blockchain provides the fine-grained data sharing and full traceability with immutable transaction data.

In summary, our UniVAC system possesses the following contributions:

1. We proposed a secure, fine-grained, and scalable privacy-preserving access control solution for the vaccine passport stored in the IPFS based on symmetric encryption and CP-ABE, and blockchain network segmentation.
2. We introduced the outsourced CP-ABE decryption as a part of proxy re-encryption process to support the secure transfer of the encrypted vaccine passport to the authorized requestors without the dependency on data owner in an optimized communication cost.
3. We provided the details of the design and implementation of the blockchain to enable the verification to be done in a decentralized way with the guarantee of traceability and immutable records of data. We also implemented the smart contracts to systematically manage user registration and flexible and secure enforcement of the consent. All consent histories are well kept in the blockchain.
4. We conducted the extensive experiments in a real cloud and blockchain to demonstrate that our UniVAC is efficient for real implementation.

This paper is organized as follows. Related work section discusses related works. Our proposed system section presents the system model and the security model of our proposed system. Security properties section provides the security analysis of our scheme. Evaluation section provides the evaluation analysis and experiments. Concluding remark section gives conclusion and future work.

Related work

Current research works related to secure data sharing in cloud environment tend to apply blockchain technology to enable the authentication and permission validation, and access transaction to be done in the decentralization way. This plaform enhances the accessibility and functionality of access control management. Liu et al. [8] proposed a data sharing framework using smart contracts and blockchain technology for tracing and enforcing agreements. The smart contracts are generated from the parameters specified in the legal data sharing protocol. Lin et al. [9] introduced a secure mutual authentication based on blockchain technology to enforce fine-grained access control policies supporting data privacy and security. In this scheme, the authors applied attributed-based signature and multi-receiver encryption and MAC technique for guaranteeing authenticity and auditability.

Wang et al. [10] proposed an access control scheme to support secure data sharing in cloud. They employed Ethereum blockchain to store key information and access transactions through smart contracts. The ciphertexts are encrypted based on the CP-ABE while the user secret key is encrypted by the AES algorithm. In this scheme, the data owner can specify the period of data access in the policy through the smart contract. Therefore, the data user can decrypt the ciphertext if and only if the time is within the valid access period.

In [11], Yang et al. proposed a blockchain-based access control framework called AuthPrivacyChain by focusing on the privacy of shared resource while user authentication and authorization function are done by the smart contracts. In this scheme, the authentication source and authorization policy are encrypted and stored in the blockchain. Even though the proposed scheme fully supports the privacy of both data and access policy, the cost for validating encrypted credentials and the policies in the block bring the performance issue.

In [16], Fan et al. introduced a secure and traceable data sharing scheme using CP-ABE. In this scheme, the authors employed CP-ABE for encrypting the data and used a private blockchain to house the key generation function for generating the secret key. In the blockchain, the data owner can verify the data users' identity and enforce the authorization based on the predefined access policy.

Yuan et al. [17] and Wu et al. [18] proposed a data privacy protection scheme to support secure and fine-grained data sharing based on CP-ABE and blockchain system. In these schemes, the blockchain stores access transactions and the access policy for controlling access permissions of users. If there are unauthorized accesses or any malicious activities occur, the system provides

audit trails to support the traceability of cryptographic operations and transaction activities.

In [12], Yazdinejad et al. proposed a decentralized authentication scheme for patients in a distributed hospital network based on blockchain. The authentication protocol was proposed to authenticate the different users in the hospital based on the public key cryptosystem. However, the paper only focuses on the authentication issue, it does not entail the secure medical data sharing and verification issue.

In [2], S. Fugkeaw introduced a blockchain-based e-KYC scheme based on the ciphertext policy attribute-based encryption (CP-ABE) method binding with the client consent. The author proposed a lightweight cryptographic protocol based on the combination of a symmetric encryption and a public key encryption to encrypt the data. CP-ABE is used to encrypt the blockchain transactions. Smart contracts are proposed to automate and control the registration, consent enforcement, and document verification process.

In [19], Dinh et al. proposed a trusted authority to perform electronic health records (EHRs) encryption and decryption. The proposed scheme aims to minimize the workload of the user to make the scheme as lightweight as possible. Their work focuses on EHRs data access over mobile devices. In their scheme, the data will be encrypted with a trusted authority public–private key pair and stored on IPFS, a cloud storage service. When the user requests the data, the trust authority will access the requested ciphertext stored in the IPFS. Then the ciphertexts are decrypted before they are sent to the user via a secure channel.

Recently, there are research works focusing on the development of secure health passport data sharing by using blockchain and encryption techniques. Hasan et al. [3] proposed a digital health passport system based on the combination of blockchain model in [4], smart contracts, and proxy re-encryption (PRE). In this system, the data owners are able to grant access to their health passport to other users has control over his data. The proposed scheme used smart contracts based on the Ethereum blockchain to store a digital medical identity for test-takers that allows trusted authorities to verify their identity and direct them to the immunity records stored in the IPFS and only their hash values are stored in the smart contracts. Basically, the data owner encrypts their immunity-related documents or vaccination passport based on the symmetric encryption. Then, the documents are uploaded to the IPFS servers. When there is an access request, the data owner generates a new re-encryption key and send it to the proxy for re-encryption. After the re-encryption process, the new key encrypted with the requestor's public key is sent to the requestor. The data requestor then decrypts the key using her private key to decrypt the encrypted symmetric key. The receiver finally uses the symmetric key to decrypt the content of the encrypted passport. Even though this scheme offers the confidentiality of the digital immunity documents, there is an overhead that the data owners need to deal with the re-encryption process. This cost will be very expensive if there are a high number of requestors.

In [20], Gao et al. proposed an immunity passport scheme based on the dual blockchain model and searchable encryption technique to support secure immunity passport data sharing and ciphertext search capability. In this scheme, the domestic blockchain and international blockchain are proposed to handle the immunity data storage and verification in the different levels. In this model, the user gets his pseudoidentity and his full public–private key pair from the key generation center (KGC). When the user gets vaccinated from the authorized agency, the agency issues an immunity passport and encrypts it by using public key and random encryption. Finally, the ciphertext is sent to store in the IPFS and its hash is stored in the blockchain. However, this scheme encounters high cost for calculating the authentication signature for every user.

In addition, some solutions [12, 21–23] were proposed to support the authentication for medical record tracing scenarios such as the smart grid, the IoTs, and the smart medical. However, these techniques have not addressed the scalable access control and secure delegation of the validation of vaccination records to trusted parties.

Our proposed system

This section presents the system model, security model, and the cryptographic construct of our UniVAC system.

System model overview

Figure 1 presents the overview of our UniVAC System Model.

UniVAC is designed to automatically verify the validity of Covid-19 vaccine passports for international travelers. Basically, the data owners who get vaccinated is asked to give the consent for allowing the epidemic prevention agencies (EPAs) to keep their vaccine record in the cloud for further access by the authorized parties. All vaccine records are encrypted based on the CP-ABE method. In our scheme, we use blockchain to store transaction records and support data indexing and develop and smart contract to perform user authentication and e-consent generation. The blockchain networks are segmented

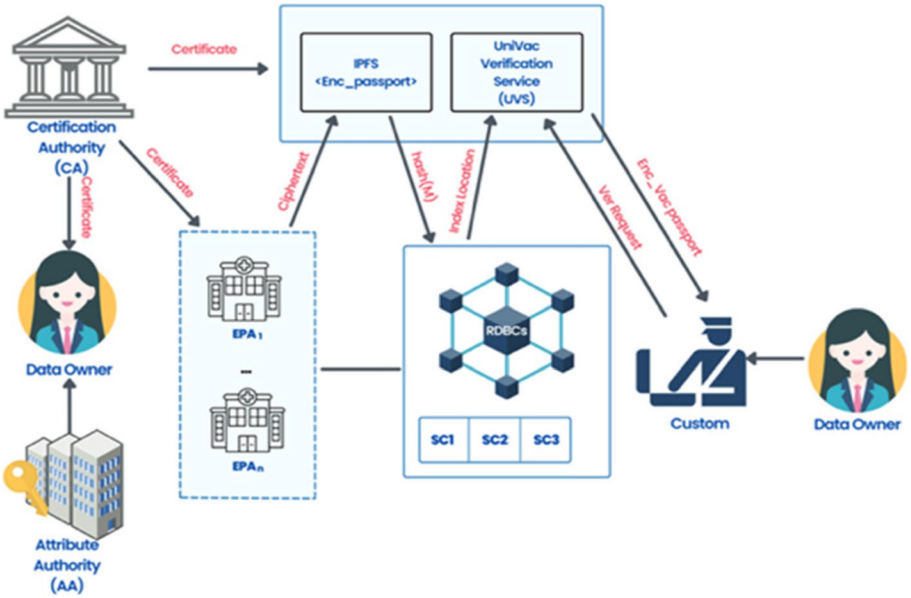

Fig. 1 UniVAC system model

based on the EPAs location. In the cloud environment, we propose a UniVAC verification service (UVS) to manage the verification service where the requests are initiated from the trusted parties such as international immigration offices. The verification is done through the proxy re-encryption mechanism using the outsourced CP-ABE decryption and PKI method.

The system model consists of the following entities: Attribute Authority (AA), Certification Authority (CA), Epidemic Prevention Agencies (EPA), Inter-Planetary File System (IPFS), Users, International Immigration offices, Regional Domestic Blockchains (RDBCs), and UniVAC Verification Service (UVS). The detail of each entity is described below.

- Attribute Authority (AA): AA is a trusted authority that generates the public parameter *PK* and the master private key *MSK* of the system. The AA stores the *MSK* and publishes *PK* available for the users. The AA also issues a CP-ABE secret key generated and securely distributed to users.
- Certification Authority (CA): CA is a trusted authority responsible for issuing a X.509 certificate to each EPA and UniVAC verification service, and international immigration officers.
- Epidemic Prevention Agencies (EPAs): The EPA creates a key pair and publishes its public key in the blockchain. EPA is authorized to vaccinate users, generate the vaccine record, and store the ciphertext of the vaccine record or the vaccine passport in the IPFS.
- Inter-Planetary File System (IPFS): IPFS is a protocol and peer-to-peer network for storing and sharing data in a distributed file system. It stores all ciphertexts of a vaccination record generated by EPAs. It leverages the modular P2P networking stack libp2p [24, 25].
- Regional Domestic Blockchains (RDBCs): RDBCs represent the domestic blockchain network divided into multiple segments based on the number of regions in the country. In our system, the RDBCs are used due to the restriction of data privacy and ease of participant control. We assume that RDBCs are private chains implemented by the responsible government unit such as Ministry of Health to control the vaccine or immunology transactions and to ensure compliance with data privacy regulations. RDBCs store the transactions made by EPAs in each region of the country.
- Smart Contracts (SCs) are programmable codes used to automate the logical function in the blockchain network. In our scheme, the smart contract is responsible for computing the hash value of the user's credential data after the users are enrolled by the EPA. It also generates the e-consent upon the successful generation of hash value.
- UniVAC Verification Service (UVS) is an autonomous system located on cloud. UVS system con-

sists of a cluster of servers that is responsible for managing RDBCs and coordinating with the host blockchain to support the access requests.
- Data owners (DOs) refers to individuals who got inoculated by the EPA. DOs provide the consent to EPA to allow their vaccine records to be published in the IPFS.
- International immigration offices are responsible for verifying the vaccine passport of the travelers. They make a request to the blockchain manager for the verification of the vaccination record of the travelers.

Basically, when individuals get vaccinated at the EPA, they become the owner of their vaccine records. They are able to delegate their right by giving the consent to the EPA to store their records in an encrypted format in the cloud and to transfer the passport to authorized parties. Then EPA will sign the record with its private key and send the ciphertexts of the record to IPFS. The IPFS generates hash of the received ciphertext to the domestic blockchain where the EPA is close to. We assume that all the epidemic prevention agencies (EPAs) jointly maintain RDBCs. To enable the efficient vaccine record verification process, the UVS located on the cloud takes the verification requests and communicates with the blockchain network to get the index of the ciphertext stored in the IPFS. Then, the ciphertext is re-encrypted by the proxy re-encryption service by using the immigration office's public key. Hereafter, a new ciphertext is delivered to the immigration office. The immigration office can then decrypt the ciphertext and verify the content of the vaccine passport. As our proxy use the public key encryption for supporting the ciphertext transformation to deliver secure data transfer to the end user, the cost of key distribution is minimized [26].

Security model

In our model, we assume that AA and CA are trusted authorities. The cloud and proxies are honest but curious. They generally perform honestly for all tasks they are delegated, but they are curious about the outsourced data. The outsourced ciphertext may be compromised based on the brute force decryption or attribute collusion. In cloud environment, the adversary can be any entities who corrupt authorities only statically, but queries can be made adaptively. The attack of the security can be done by an adversary requesting a secret key from the attribute authority.

Since the vaccine record is encrypted with the CP-ABE method, it is proven to be secure against collusion attack. The detailed proof of cryptographic strength can be found in [6, 27]. The security model of our scheme is defined as a game-based in compromising the CP-ABE key to obtain the capability in accessing the plaintext. The game-based between an adversary A and a challenger C is defined as follows:

Setup. For uncorrupted authorities AA, the challenger C runs CreateAttributeAuthority algorithm and sends a public keys PK to the adversary A. For corrupted authorities $AA^{'}$ the challenger sends both the public key PK and the secret key SK to adversary A.

Phase1: The adversary A delivers SK *which* is a set of attributes issued by an uncorrupted authority AAk. The challenger C gives the secret key SK to the adversary A.

Challenge. Adversary A sends two challenge messages $m1$ and $m2$ to the simulator. The simulator flips a fair binary coin ν, and returns an encryption of $m\nu$. In this game, the CT_{VR_id} which is a ciphertext of the vaccine passport encrypted by a CP-ABE method. The ciphertext CT_{VR_id} is computed as follows:

$CT_{VR\_id}=(T, \hat{C} = \mathrm{m}_{\nu}z, CT_k = h^s, \forall y \in Y : C_y = g^{q_y}(0), C\prime_y = H(att(y))^{q_{y(0)}})$ where γ is a chosen set of attributes. If μ = 0 then $z = e(g,g)^{\alpha s}$. Therefore, the ciphertext CT_K is a valid random encryption of message m_{ν}.

Otherwise, if $\mu = 1$ then $z = e(g,g)^{z}$. We now have, $\hat{C} = m_{\nu}\, e(g,g)^{z}$. Since z is random, $\hat{C}$ will be a random element of G_1 from the adversaries view and the message contains no information about m_{ν}.

Phase 2. The simulator does as it did in Phase 1.

Guess Adversary A sends a guess of ν' *of* ν.

The advantage of A in this game is defined as:

$$ADV_A = \Pr\left[\nu = \nu'\right] - {}^{1}\!/_{2}.$$

Definition 3: Our proposed scheme is secure if all polynomial time adversaries have at most a negligible advantage in the above game.

Theorem 1: *Suppose there is no polytime adversary who can break the security of CP-ABE with nonneglible advantage; then there is no polytime adversary who can break our crypto system with nonnegligible advantage.*

Proof As we have shown how the adversary A has nonnegligible advantage against our scheme. Similar to A, we show how the adversary B, is created to break the CP-ABE scheme with nonnegligible advantage. The adversary B can play a similar game with the CP-ABE scheme to make private queries during the game to get the private keys in the CP-ABE scheme.

Initialization. The adversary B takes the PK of the authority k, $PK'_k = \{G_0, g, h = g^{\beta}, f = g^{\frac{1}{\beta}}, e(g,g)^{\alpha}\}$, and the corresponding secret key (β, g^{α}). is unknown to the adversary.

Setup. The adversary B gets the public parameters from *PK'* as $PK_k = \{G_0, g, h = g^{\beta}, f = g^{\frac{1}{\beta}}, e(g,g)^{\alpha}\}$, then the public key PK_k is sent to the adversary.

Phase 1. *B* answers private key queries. Suppose the adversary is given a secret key query for a set of attributes *S* where *S* does not satisfy *T*. Here, *B* makes a query for obtaining *SK* for the same set *S* twice. Then, *B* obtains two different *SKs* as follows.

$$SK_k = (D = g^{(\alpha_k + r)/\beta_k}, A_i \in S : D_i = g^{r}.H(i)^{r_i}, D'_i = g^{r_i}).$$

$$SK'_k = (D = g^{(\alpha_k + r')/\beta_k}, A_i \in S : D_i = g^{r'}.H(i)^{r'_i}, D'_i = g^{r_i}).$$

where *i*'s are attributes from *S*, and *r, r',r_i, r_i'* are random number in Z_p. With SK_k and SK'_k, *B* can obtain $g^{r-r'/\beta}$, and chooses random number t_i, $t_{i,j} \in Z_p$. Let $r^* = t_i - r_i$ and $r'' = t_{i,j} - r_i'$. Then *B* derives the SK requested by *A* as $SK^* = (D = g^{(\alpha_k + r')/\beta_k}, D = g^{(\alpha_k + r)/\beta_k}, A_i \in S: D_i = g^{r^*}.H(i)^{r''_i}, D'_i = g^{r''_i})$. Then, the *SK* is returned to the adversary *A*.

Challenge. When *A* decides that Phase 1 is over, it outputs an access policy *T* and two messages m_1 and m_2, which it wishes to be challenged. *B* gives the two messages to the challenger, and is given the challenge ciphertext CT_K. Then *B* computes the challenges ciphertext for *A* from CT_K as $CT_K{}^*$. Finally, the challenge ciphertext $CT_K{}^*$ is returned to the adversary *A*.

Phase 2. *A* makes queries not issued in Phase 1. *B* responds as in Phase 1.

Guess. Finally, it outputs a guess $v' \in \{1,0\}$, and then *B* concludes its own game by generating *v'*. According to the above security model, the advantage of the adversary *B* is:

$$ADV_A = |\Pr[v = v'] - {}^1\!/_2| = ADV_B$$

Thus, *B* has nonnegligible advantage against the CP-ABE, which completes the proof of the theorem.

Enrollment

In our system, the users or the vaccine passport owners need to enroll to the system since they get the vaccination. First, the system verifies the identity of each user. Here, the EPA connects to the blockchain and runs the smart contract. The smart contract records the hash value of IDs and public key of the user for future matching with the corresponding ciphertext and user authentication. Simultaneously, the system asks users to submit their public key together their citizen or passport ID. Then, the smart contract generates the consent to ask the users (vaccine record owners) to allow the EPA to store their vaccination records in the cloud storage as well as to transfer their records to trusted parties.

```
Def CreateConsent(UserID, ConsentContent):
      if Consent==accept then
      DigitalSignature = ConsentSign(Consent,PrivK_do_id)
      Store consent<consent,DigitalSignature>
      End if
      if consent==deny then
      throw
      End if
End Def
Def Enroll(UsersCitizenID, PubK_do_id):
   h = hash(CitizenId);
   Store credential<h,PubK_do_id>
   CreateConsent(UserID, ConsentContent)
End Def

Enroll(UsersCitizenID, PubK_do_id))
```

Algorithm 1. User enrollment and consent generation smart contract

In our system, we assume that the user needs to have a key pair and submit the public key in the enrollment phase. The blockchain does not retain any plaintext of user identity and the smart contract can be only activated by the authorized EPA.

Cryptographic construct

This section presents the cryptographic construct of our UniVAC system. Table 1 presents the notations and their meaning used in our paper.

A concrete construction of UniVAC system consists of a set of algorithms organized in six phases as follows.

Phase 1: system setup

This phase is run by the AA to set up the system and security parameters necessary for supporting CP-ABE cryptographic process.

- CreateAttributeAuthority(k)PK_k, SK_k, $PK_{x,k}$. This algorithm uses an attribute authority ID(k) and selects a bilinear group G_0 of prime order p with generator g. Then, it selects two randoms α, $\beta \in Z_p$*to* compute the public key defined as follows:

$$PK_k = \left\{G_0, g, h = g^{\beta}, f = g^{\frac{1}{\beta}}, e(g,g)^{\alpha}\right\}.$$

Then, the authority computes the secret key SK_k as (β, g^{α}).

Table 1 Notations and its meaning

Notation	Meaning
S_k	Set of all attributes issued by authority *k*
S_{DO_id}	Set of all attributes issued to data owner *DO_id*
SK_{DO_id}	A secret key which belongs to data owner *DO_id*
SK_{UVS}	A secret key issued to UVS
MSK_k	Master secret key which belongs to authority *k*
PK_k	Public key which belongs to authority *k*
$PubKey_{immigration\_id}$	A public key which belongs to Immigration office id
$PubKey_{do\_id}$	A data owner's public key
$PrivKey_{do\_id}$	A data owner's private key
$(PubKey_{EPA\_id}, PrivKey_{EPA\_id})$	A RSA key pair which belongs to *EPA id*
VR_id	A vaccine record id
$MD\_CT_{VR\_id'}$	A message digest or hash value of the ciphertext of vaccine record CT_{VR_id}
SymKey	An AES symmetric key created to encrypt the initial vaccine record
ACP	An access control policy used to encrypt the data based on CP-ABE method
CT_{VR_id}	An encrypted vaccine record

Phase 2: key generation

This phase provides cryptographic keys consisting of RSA key pair and CP-ABE key to users and system entities.

- GenerateRSAKeyPair(RSAKeyGenFunction)($PubK_{EPA\_id}$, $PrivK_{EPA\_id}$). Each EPA runs RSAKeyGen function to generate a RSA key pair. The EPAs' public keys are published in the blockchain.
- DOKeyGen(S_{DO_id}, MSK_k)SK_{DO_id}. AA takes as input a set of attributes (S_{DO_id}) identifying the *DO_id*'s decryption key, AA's public key (PK_k). For each DO *DO_id*, the *AA* chooses a random *r* and $r_j \in Z_p$, for each attribute $j \in S$. Then the DO decryption key (SK_{DO_id}) is computed as:

$$SK_{j,k} = (D = g^{(\alpha_k + r)/\beta_k}, D = g^{(\alpha_k + r)/\beta_k}, D = g^{(\alpha_k + r)/\beta_k}, A_i \in S_k : D_i = g^r.H(i)^{r_i}, D'_i = g^{r_i}).$$

This key generation process is also applied for generating SK_{UVS}.

Phase 3: encryption

This phase describes how the data encryption step is performed in our system. In our scheme, data encryption is based on CP-ABE encryption because it provides fine-grained access control and one-to-many encryption. Data owners and UVS can use the secret key or a CP-ABE decryption key to decrypt the encrypted vaccine record. For UVS, it uses the secret key (CP-ABE key) to support re-encryption process. Therefore, our scheme allows the decryption to be performed by both data owners (vaccine record's owners) and the proxy. The proxy decrypts it upon the request from authorized partiesThe detail of the algorithm is described as follows:

$$\mathrm{ENC}(PK_k, VR\_id, R, ACP)(CT_{VR\_id'}).$$

The encryption is done by the EPA staff to call the encryption functions which performs the following steps.

- Encrypt Vaccine Record *VR_id*: the algorithm takes as inputs authority public key PK_k, *ACP*, and *VR_id*. In our model, the ACP consists of the set of attributes < DO_id, EPA_id, and UVS_id >. Then, the encrypted vaccine passport CT_{VR_id} is produced by the following encryption function:

$$VR\_id \mapsto ENC_{CP-ABE}(PK_k, ACP, VR\_id) \equiv CT_{VR\_id}$$

Then, CT_{VR_id} is sent to store in the IPFS on cloud.

- Hash CT_{VR_id}, The function takes CT_{VR_id} and SHA-256 algorithm to hash the ciphertext of vaccine record. The function is defined as:

$$CT_{VR\_id'} \mapsto hash(CT_{VR\_id}) \equiv MD\_CT_{VR\_id}$$

The message digest of ciphertext $MD\_CT_{VR\_id}$ is then produced and it is stored in the blockchain as an index tuple < *EPA_id, DO_id, VR_id,* $MD\_CT_{VR\_id}$ >.

Phase 4: index and ciphertext retrieval

This phase is for retrieving the index and the ciphertext of the vaccine record when the users authenticate

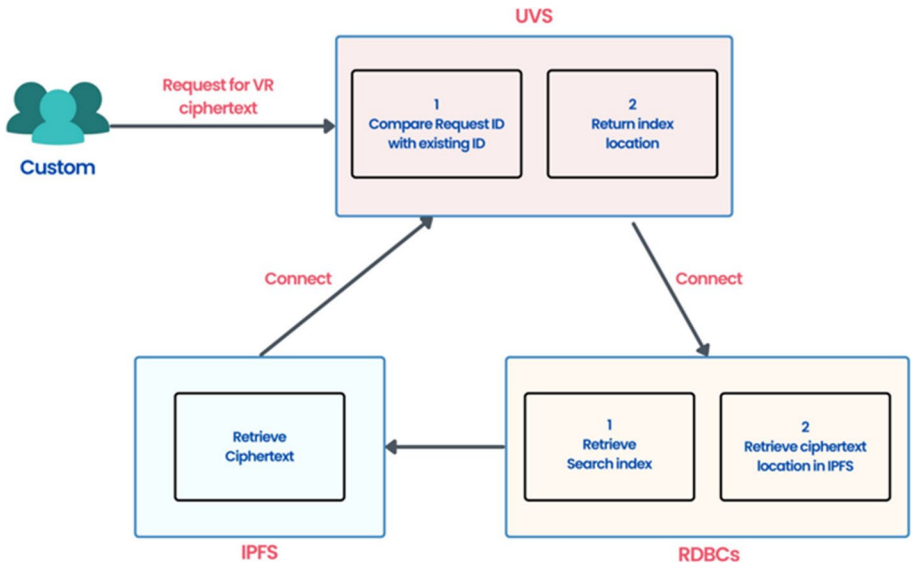

Fig. 2 Index and Ciphertext Retrieval diagram

themself at the foreign immigration office. Figure 2 presents simplified view of how the ciphertext is retrieved.

Basically, the immigration officer makes a request to the UVS for checking the user's vaccine status. UVS then checks the VR_id and gets the index tuple from the blockchain. The algorithm below shows how the indexing data is invoked and the ciphertext is retrieved based on multi-thread processing.

```
Def vacVerify(Vac_requestId):
        RDBCs = blockchain.connect()
        index = blockchain.search(Vac_requestID). getIndex()
if index != null then
        ipfs = IPFS.connect(index)
        CT_VR = ipfs.retriveCT(index)
        return CT_VR
End if
sent CT_VR to UVS
End Def
maxThreadPool = Total number of threads available
D = Duration
Num = Number of requests received by the system
thread = new Thread(maxThreadPool)
Loop Num!= 0:
        if thread.available then
                thread.setTimer(D)
                thread.execute(vacVerify(requestID, storeId))
                Num = Num - 1
        End if
End Loop
```

Algorithm 2. Retrieve index and ciphertext

After the respective encrypted vaccine record is retrieved, the UVS runs a proxy re-encryption function as described in the next phase.

Phase 5: proxy re-encryption

This phase is run by the proxy for re-encrypting the vaccine records based on the requester's public key encryption before they are sent to the requester such as the immigration office. This phase is omitted if the decryption is done by the data owner as she has the CP-ABE decryption key to decrypt the ciphertext. The re-encryption algorithm conducted by the proxy is detailed as follows.

$$\text{ReENC}(SK_{UVS}, CT_{VR\_id}, SK_{UVS}, PubKey_{immigration\_id}) \equiv \text{Enc\_}CT_{VR\_id}.$$

The proxy performs re-encryption by executing the following functions.

(1) Decrypt $CT_{VR\_id}(SK_{UVS}, CT_{VR\_id})$ *VR_id.* This function takes as inputs the UVS's secret key and the encrypted vaccine record CT_{VR_id}. It outputs the *VR_id*.

(2) ReENC*VR_id*($PubKey_{immigration\_id}$,*VR_id) Enc_VR_id* This function takes an inputs immigration office's public key $PubKey_{immigration\_id}$ and the vaccine record *VR_id.* The encryption algorithm is based on RSA algorithm using 2048-bit key size. Finally, it outputs the final encrypted vaccine record *Enc_VR_id.* The re-encryption function is defined as:

$$VR\_id \mapsto ENC_{RSA}(PubKey_{immigration\_id}, VR\_id) \equiv Enc\_VR\_id$$

Then, UVS sends the *Enc_VR_id* to the international immigration who requests for the verification of the vaccine record of the traveler.

Phase 6: verification

This phase is to verify the vaccine record content through the decryption. In case of the verification request is initiated by the authorized party such as the immigration office, the public key decryption is done as follows.

- Decrypt$Enc_VR_id(Enc_VR_id, PriVKey_{immigration_i})$: The immigration officer runs the decryption function which takes inputs immigration's office' private and the encrypted vaccine record. Finally, it outputs the vaccine record that can be used for the verification. The RSA decryption is defined as follows.

$$Enc\_VR\_id \mapsto DEC_{RSA}(PriVKey_{immigration\_id}, Enc\_VR\_id) \equiv VR\_id$$

Security properties

Our proposed scheme achieves the following security properties.

Privacy-preserving vaccine records with data owner's consent

Since our core cryptographic protocol for protecting the content of vaccine record is based on the CP-ABE method, the cryptographic strength is proven secure [6]. Also, our system generates an e-consent to satisfy privacy regulations in healthcare data privacy compliance.

Trust of proxy re-encryption

Even though the proxy is semi-trusted server, it is installed with the X.509 certificate. It is designed to run the function when it gets the legitimate request from the authorized entity based on the verification of the public key certificate. Therefore, no fake entity can execute the re-encryption function unless it has a valid certificate issued by the certification authority.

Traceability

All access activities initiated by EPAs, data owners, or international immigration offices are well recorded by the blockchain. All records are tamper-proof and chronologically ordered. Data owners or any third-party auditors can trace who performed the activities or accessed the locally stored vaccine records.

Evaluation

This section describes the evaluation of our proposed scheme through the comparative analysis and experiments.

Functional analysis

We compare the functional system between our proposed scheme, scheme [3] and scheme [20] which focused on the design and development protocol to support vaccine passport verification using blockchain. Table 2 exhibits the functional comparisons of three schemes.

As shown in Table 2, all schemes leveraged the cloud storage such as IPFS to store encrypted vaccine passport and used blockchain to support user authentication and store access transactions of which their integrity is well preserved by the hashing mechanism of the blockchain network.

Regarding the consent enforcement, only our scheme provides the e-consent through the smart contract and the consent is securely stored in the blockchain. For the capability in supporting scalability of ciphertext search, scheme [3] did not provide the indexing method. Scheme [20] generates a trapdoor to support the ciphertext search while our scheme used the hash-based indexing with the proposed segmentation of multiple blockchains to store the regional vaccine records created by the EPAs in the country. Therefore, it minimizes the cost of index search to all ciphertexts in the centralized source in the blockchain.

Finally, only our scheme applies CP-ABE to enable the fine-grained access control to the shared data. Our scheme also supports the delegation of data access based on the requests from international immigration offices by using proxy re-encryption technique.

Computation cost analysis

We compare the computation cost of our proposed scheme with two related works: [3] and [20]. To provide more understanding about the analysis, we use the following notations to describe the computation cost of all schemes.

Table 2 Comparison of System Functions

Scheme	Cloud -Block chain	Client Consent	Provision of scalable ciphertext retrieval	Fine-grained access control and delegation-based
[3]	√	X	X	X
[20]	√	X	√	X
Ours	√	√	√	√

C_e: Exponentiation cost
C_p: Pairing operation cost
C_m: Multiplication operation cost
C_h: Processing time for hashing
$|S|$: The size of user attribute set
$|Att|$: The number of attributes satisfying the policy.
SymEnc/Dec: Symmetric Encryption/Decryption cost based on 256-AES
PubEnc/Dec: Public key encryption cost based on 1024-bit RSA

Table 3 presents the comparison of the computation cost of our scheme, scheme [3], and scheme [20].

To compare the computation cost of related works and our scheme, we focus on analyzing the encryption cost that happens at the data owner and cloud and the decryption cost at the user side.

In [3], the authors applied a symmetric encryption to encrypt the vaccine records. Here, the data owner needs to generate a symmetric key to encrypt her own vaccine record before it is uploaded to the IPFS. When the authorized party wants to verify the vaccination records, the data owner needs to run the proxy re-encryption function to re-encrypt the encrypted vaccination record with the requestor's public key. The user needs to use their private key to decrypt the ciphertext. However, this scheme relies on the availability of data owners to support the verification.

In [20], the authors use hashing and bilinear pairing method to generate the ciphertext of the vaccination passport together the creation of trapdoor for indexing. In addition to encrypting the vaccine passport, the data owner uses a session key to encrypt the decryption key and sends it to the immigration staff. To access the ciphertext, the immigration staff calculates the session key to access the decryption key and uses a trapdoor broadcasted in the blockchain for searching the ciphertext. The complexity of this scheme is high as the cost of hashing, pairing, and multiplication are required. This approach also relies on the data owner to encrypt and send the decryption key to the requestors upon the access requests. This is not practical for large-scale implementation. In our scheme, we used a CP-ABE method to encrypt the vaccine passport. We did not use a symmetric encryption because the size of vaccine passport is small, and the policy used to encrypt the data is deterministic based on a few set of identity attributes of EPA, user, and UVS. To this end, our scheme does not deal with handling the cost of key encryption. With the use of CP-ABE based access policy, the scheme provides flexibility to authorized parties to access the ciphertext without the assistance of the data owner. Also, our scheme outsources re-encryption cost to the proxy for performing public key encryption while the international immigration staffs can decrypt the encrypted vaccine passport by using their private key.

Performance evaluation

For the performance analysis, we conducted the experiments to measure the ciphertext retrieval time and compare encryption and decryption time between our proposed scheme and related works [3] and [20]. For the experimental setting, we use Open SSL as a core PKI service to generate key pairs to users and proxy in our system. The core CP-ABE toolkit (https://acsc.cs.utexas.edu/cpabe/) and Java Pairing-Based Cryptography [28] for constructing the crypto service used in all schemes. We conducted the test on the CP-ABE container run on the Microsoft Azure Cloud 1 vCPU, RAM 1024 MB for the proxy side, and MacBook Pro 2018, 2.3 GHz Quad-Core Intel Core i5, RAM 8 GB for the user side.

The major aim of our evaluation is to evaluate the efficiency and practicality of our proposed cryptographic-based access control and ciphertext retrieval time with our proposed design of blockchain network. Since we used the standard cryptographic methods with no technical restriction on the proprietary blockchain operations, our scheme can be implemented in any platforms supporting blockchain operations. However, we specifically chose Hyperledger for our simulation since it is a permissioned blockchain that is suitable for storing healthcare-related data or transactions. In Hyperledger platform, the authorized participants/users have to be predefined. Hence, the access to the network is restricted only to them. To this end, our simulation was done on a local Fabric network consisting of a channel with one ordering node, two-organizations with two peers. All

Table 3 Comparison of Computation Cost

Scheme	Encryption cost	Re-encryption cost	Decryption Cost
[3]	$SymEnc + PubEnc$	$PubDec + PubEnc + SymEnc$	$SymDec + PubDec$
[20]	$4C_h + 4C_e + 4C_m + C_p$	-	$C_e + 5C_m + C_p\ C_h$
Ours	$(2\|N\|+3)\ C_e$	$(2\|Att\|+1)C_p + (\mathbf{2}\|N\|+\mathbf{2})\ C_e + PubEnc$	$PubDec$

system nodes (e.g., order, peer) were implemented in docker containers run on Intel Xeon 3.4 GHz with 16-GB RAM. The machine was installed with Ubuntu 16.04 LTS with Fabric V1.4.

In the experiment, we used a fixed size of vaccine passport file which is 20 KB and used the access policy containing 5 attributes and 10 attributes contain in the user secret key for the test. Blockchain network houses 10,000 blocks and the size of each block is 40 KB. In our experiments, we run the test 100 times for all performance test and used the average time to plot the graphs.

- Ciphertext retrieval time

As can be seen from Fig. 3, the ciphertext retrieval time is proportional to the number of the requests and the size of blockchain. Our scheme delivered smallest ciphertext retrieval time than [3] and [20]. This is because of the segmentation of multiple blockchains that store the regional vaccination records and multi-thread processing. The advantage of our proposed ciphertext retrieval is more obvious when there are a larger volume of access requests. Scheme [3] took highest time as it required the data owner to execute the re-encryption function in the ciphertext retrieval process.

To demonstrate the performance of cryptographic operations, we also did the experiments to measure the encryption and decryption time of [3, 20] and our scheme.

- Encryption/Decryption Performance

As presented in Fig. 4, scheme [3] took highest time as it required the data owner to runs two encryption algorithms, public key encryption and symmetric key encryption, while scheme [20] uses hash operation, exponential operation, paring operation and multiplication operation for the encrypting process, the encryption cost of [20] is close to our scheme since both schemes relied on the exponential operation.

For the decryption time, as can be seen in Fig. 5, our scheme took least decryption time because our scheme only used public key decryption algorithm while scheme [20] required hash operation, exponential operation, pairing operation and multiplication operation for decrypting process. These operations took higher computation cost than the public key decryption alone. For [3], it took highest decryption time as it required the user to run two decryption algorithms, the public key decryption algorithm and the symmetric key decryption algorithm to decrypt the vaccine passport data.

Concluding remark

In this paper, we have proposed the secure and scalable access control for outsourced vaccination records based on the blockchain technology, CP-ABE, and proxy re-encryption. We introduced a well-established design and implementation of our blockchain segmentation scheme and cryptographic model to deliver practical vaccination record validation requested by trusted parties such as international immigration offices. Our scheme also provides flexibility for rendering the vaccination status to the vaccination record owners based on the access policy enforced over the ciphertext. We conducted the experiments using Azure cloud and Hyperledger blockchain to substantiate that our proposed scheme is practical and efficient in providing comparable encryption time and

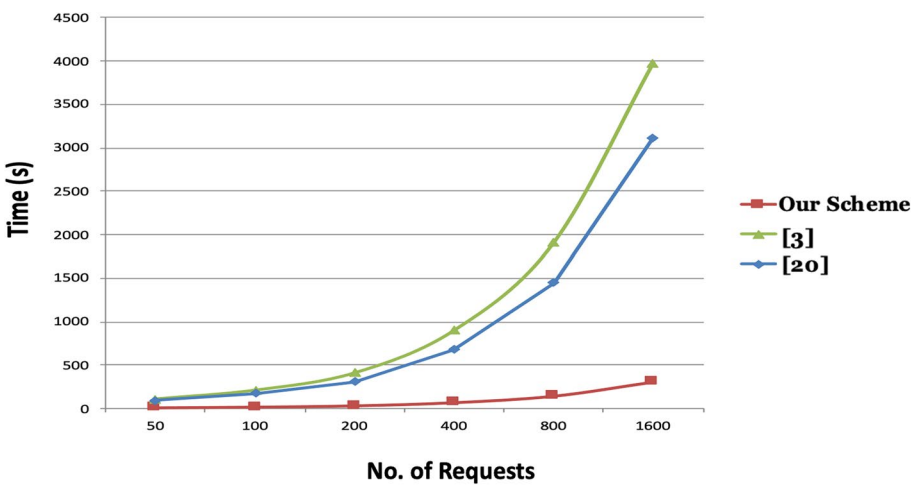

Fig. 3 Ciphertext Retrieval Time

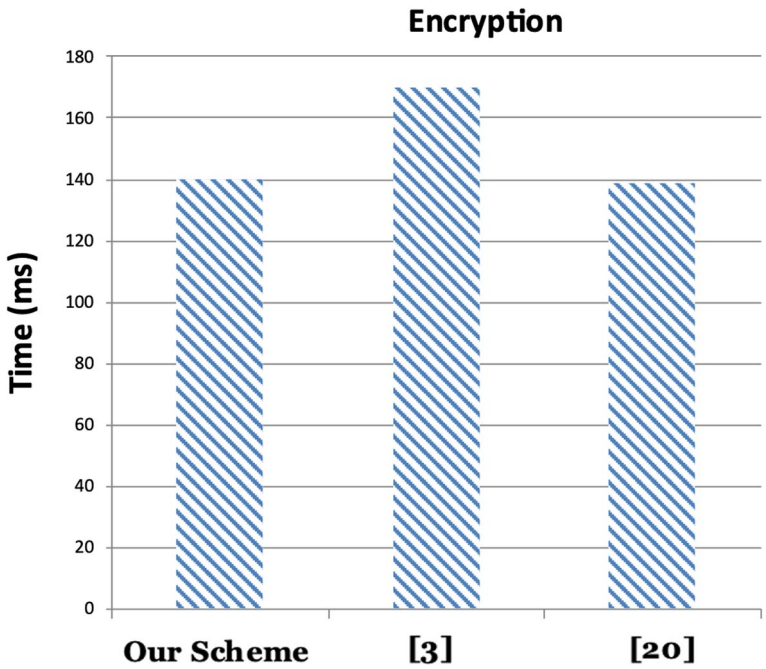

Fig. 4 Encryption Time

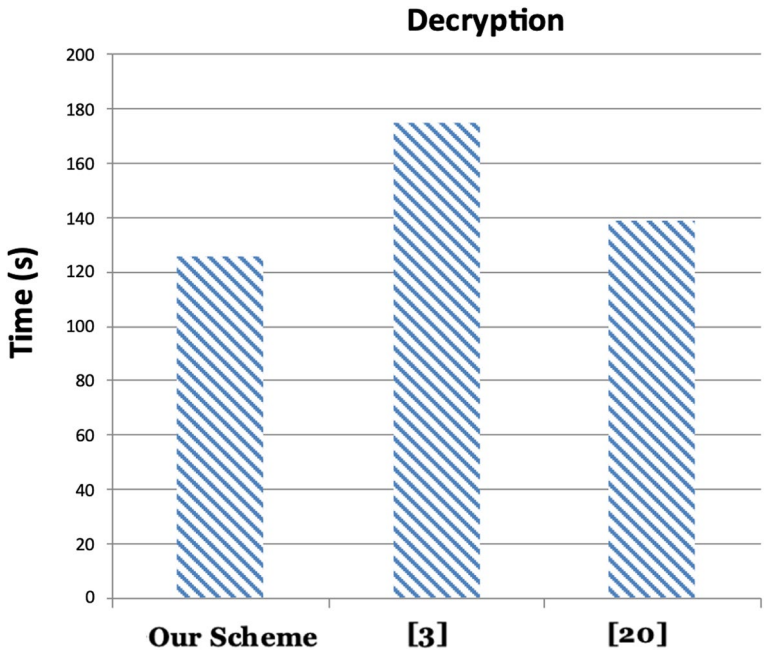

Fig. 5 Decryption Time

decryption time as the state of the arts and giving higher efficiency for accommodating a large volume of vaccination status validation requests.

For future works, we will tackle the lightweight protocol for searchable encryption and public auditing of the vaccination records stored in the cloud.

Author's contributions
This paper is written by Somchart Fugkeaw.

Funding
The authors received no specifc funding for this study.

Availability of data and materials
Not applicable.

Declarations

Competing interests
The authors declare no competing interests.

Received: 27 December 2022 Accepted: 10 July 2023
Published online: 02 August 2023

References

1. Jackson EB, Dreyling R, Pappel I (2021) Challenges and Implications of the WHO's Digital Cross-Border COVID-19 Vaccine Passport Recognition Pilot. Eighth Int Conference eDemocracy eGovernment (ICEDEG) 2021:88–94. https://doi.org/10.1109/ICEDEG52154.2021.9530954
2. Fugkeaw S (2022) Enabling Trust and Privacy-Preserving e-KYC System Using Blockchain. IEEE Access 10:49028–49039
3. Hasan HR, Salah K, Jayaraman R et al (2020) Blockchain-based solution for COVID-19 digital medical passports and immunity certificates. IEEE Access 8:222093–222108
4. Chaudhari S, Clear M, Bradish P, Tewari H (2021) Framework for a DLT based COVID-19 passport. Intelligent Computing 108–123
5. Khan D, Hashmani MA, Jung LT, Junejo AZ (2022) Blockchain Enabled Track-and-Trace Framework for Covid-19 Immunity Certificate. 2022 2nd International Conference on Computing and Information Technology (ICCIT) 2022:248–253, 2022.
6. Bethencourt J, Sahai A, Waters B (2007) Ciphertext-policy Attribute-based Encryption. IEEE Symposium of Security and privacy, Oakland, CA, USA
7. Fugkeaw S, Sato H (2015) An Extended CP-ABE Based Access Control Model for Data Outsourced in the Cloud. 2015 IEEE 39th Annual Computer Software and Applications Conference 73–78, https://doi.org/10.1109/COMPSAC.2015.216.
8. Liu K, Desai H, Kagal L (2018) Enforceable data sharing agreements using smart contracts.arXiv:1804.10645. Available: https://arxiv.org/abs/1804.10645
9. Lin C, He D, Huang X, Choo K-KR, Vasilakos AV (2018) BSeIn: A blockchain-based secure mutual authentication with fine-grained access control system for industry 40'. J Netw Comput Appl. 116:42–52
10. Wang S, Wang X, Zhang Y (2019) A Secure Cloud Storage Framework With Access Control Based on Blockchain. IEEE Access 7:112713–112725. https://doi.org/10.1109/ACCESS.2019.2929205
11. Yang C, Tan L, Shi N, Xu B, Cao Y, Yu K (2020) AuthPrivacyChain: A Blockchain-Based Access Control Framework With Privacy Protection in Cloud. IEEE Access 8:70604–70615. https://doi.org/10.1109/ACCESS.2020.2985762
12. Yazdinejad A, Srivastava G, Parizi RM, Dehghantanha A, Choo KKR, Aledhari M (2020) Decentralized authentication of distributed patients in hospital networks using blockchain. IEEE J Biomed Health Inform 24(8):2146–2156
13. Ullah Z, Raza B, Shah H, Khan S, Waheed A (2022) Towards Blockchain-Based Secure Storage and Trusted Data Sharing Scheme for IoT Environment. IEEE Access 10:36978–36994. https://doi.org/10.1109/ACCESS.2022.3164081
14. Li X, Liu S, Lu R, Khan MK, Gu K, Zhang X (2022) An Efficient Privacy-Preserving Public Auditing Protocol for Cloud-Based Medical Storage System. IEEE J Biomed Health Inform 26(5):2020–2031. https://doi.org/10.1109/JBHI.2022.3140831
15. Fugkeaw S (2023) Achieving Decentralized and Dynamic SSO-Identity Access Management System for Multi-Application Outsourced in Cloud. IEEE Access 11:25480–25491. https://doi.org/10.1109/ACCESS.2023.3255885
16. Fan Y, Lin X, Liang W, Wang J, Tan G, Lei X, Jing L (Accepted/In press). TraceChain: A blockchain-based scheme to protect data confidentiality and traceability. Softw Pract Exp. https://doi.org/10.1002/spe.2753
17. Yuan C, Xu M, Si X, Li B (2017) Blockchain with accountable CP-ABE: How to effectively protect the electronic documents, in Proc. IEEE 23rd Int Conf Parallel Distrib Syst 800–803.
18. Wu A, Zhang Y, Zheng X, Guo R, Zhao Q, Zheng D (2019) Efficient and privacy-preserving traceable attribute-based encryption in blockchain. Ann Telecommun 74(7–8):401–411
19. Nguyen DC, Pathirana PN, Ding M, Seneviratne A (2019) Blockchain for Secure EHRs Sharing of Mobile Cloud Based E-Health Systems. IEEE Access 7:66792–66806. https://doi.org/10.1109/ACCESS.2019.2917555
20. Gao H, Ji H, Huang H, Xiao F, Jian L (2022) An Immunity Passport Scheme Based on the Dual-Blockchain Architecture for International Travel. Wireless Commun Mobile Comput. 2022:1–1. https://doi.org/10.1155/2022/5721212
21. Xu H, Zhang L, Onireti O, Fang Y, Buchanan WJ, Imran MA (2021) BeepTrace: blockchain-enabled privacy-preserving contact tracing for COVID-19 pandemic and beyond. IEEE Internet Things J 8(5):3915–3929
22. Garg L, Chukwu E, Nasser N, Chakraborty C, Garg G (2020) Anonymity preserving IoT-based COVID-19 and other infectious disease contact tracing model. IEEE Access 8:159402–159414
23. Jia X, He D, Kumar N, Choo K-KR (2020) Aprovably secure and efficient identity-based anonymous authentication scheme for mobile edge computing. IEEE Syst J 14(1):560–571
24. Daniel E, Tschorsch F (2022) IPFS and Friends: A Qualitative Comparison of Next Generation Peer-to-Peer Data Networks. IEEE Commun Surv Tutorials 24(1):31–52. https://doi.org/10.1109/COMST.2022.3143147
25. "libp2p—Github," Protocol Labs. https://github.com/libp2p. Accessed June. 2022.
26. Fugkeaw S (2023) Secure Data Sharing With Efficient Key Update for Industrial Cloud-Based Access Control. IEEE Trans Serv Comput. 16(1):575–587. https://doi.org/10.1109/TSC.2021.3110828
27. Cheuung L, Newport C (2007) Provable Secure Ciphertext Policy ABE. In Proc. of ACM Conference on Computer and Communication Security. pp. 456–465.
28. PBC (Pairing-Based Cryptography) library. Accessed: 2 June 2022. Available: https://crypto.stanford.edu/pbc/

Publisher's Note