A Blockchain-Based Access Control Scheme for Smart Grids

Yuyang Zhou
School of Computer Science and Engineering
University of Electronic Science and Technology of China
Chengdu 611731, China
yuyang.zhou@std.uestc.edu.cn

Yuanfeng Guan
SI-TECH Information Technology Co., Ltd
Beijing 100031, China

Zhiwei Zhang
SI-TECH Information Technology Co., Ltd
Beijing 100031, China

Fagen Li
School of Computer Science and Engineering
University of Electronic Science and Technology of China
Chengdu 611731, China
fagenli@uestc.edu.cn

***Abstract*—At present, the access control schemes in the power grid are centralized. In the centralized system, the data of the network sensor nodes is transmitted by centralized nodes, and the data itself may be illegally tamped with or lost, which can lead to reduced system reliability. For this feature, we apply blockchain technology to the design of access control schemes. In this paper, we propose a blockchain-based access control scheme that is suitable for multiple scenarios in the smart grid. Our access control scheme is based on an identity-based combined encryption, signature and signcryption scheme. In addition, we design a consensus algorithm in the power system for the consortium blockchain architecture to solve the key escrow problem of the untrusted third parties. Our scheme also ensures the confidentiality, integrity, authentication and non-repudiation of the data. Compared with the existing work, our scheme can use the same key pair to encrypt, sign and signcrypt the message, which has lower computation and communication costs in multiple scenarios of smart grids.**

***Keywords*—smart grids, access control, blockchain, combined public key scheme**

I. Introduction

The smart grid, also known as the next generation power grid, is based on the physical grid system using advanced information communication technology, sensor measurement technology, computer technology and control technology [1]. Relying on modern information technologies, the smart grid can digitally manage power production, power transmission, power division and power control. Different from the traditional power grid, a main feature of the smart grid is the ability of the two-way flow of information between the user and the power provider. For example, in a traditional power grid, electricity is generated in a power plant and then be transmitted to users through a transmission network, a branch network. But in a smart grid, electricity can also be returned to the power provider by user (e.g., user can generate electricity from their home solar panels and transfer them to the power provider). User models in smart grids can be divided into three types: Home Area Network (HAN), Building Area Network (BAN), Industrial Area Network (IAN) [2].

If a attacker illegally obtains the users power usage information, he can infer the users specific activity information according to the users power usage pattern. At the same time, if the attacker impersonates a legitimate user to transmit malicious information to the power provider (e.g., DDOS attack), it will also hinder the power providers daily work. In order to protect these sensitive information being attacked and utilized, research on access control of smart grids was proposed.

Access control ensures only authorized user can access the specified data and solves the problem of unauthorized access of important information. Access and authentication measures in existing smart grids have the following weaknesses: (1)current mainstream access control scheme implements cross-domain access through centralized authentication or third party centralized authentication, but whether a third party is absolutely credible; (2)at the same time, there are massive user access nodes and two-way information circulation in the smart grid. All of this pose a challenge to the design of access control schemes in smart grids.

Blockchain is an emerging decentralized architecture and distributed computing paradigm [3]. Blockchain technology has the characteristics of decentralization, collective maintenance, security and credibility. At present, many access control schemes adopt centralized management. If the blockchain technology is used for upgrade the present access control schemes, the traditional access control method will have the characteristics of decentralization and high reliability of the blockchain. So its especially suitable for smart grid systems, which have multiple nodes.

A. Related Work

Access control in smart grids has received much attention in recent years. In 2011, Sankar et al. [4] presented a centralized access scheme for power grids that requires the regional transmission organizations (RTOs) to be online during data transmission. However, such method can easily become a system bottleneck. Sun et al. [5] proposed an identity-based

encryption (ABE) access control scheme in smart grids, which alleviated the computational overhead of intelligent terminals. However, in [5], the master authentication center and each terminal of the jurisdiction share the key, which easily suffer from main-in-the-middle attacks. So, the confidentiality of the data cannot be guaranteed. In 2014, Wu et al. [6] proposed a lattic based access control scheme which used identity-based cryptography (IBC). However, it assumes there is a fully trusted network controller who is in charge of the whole network. In 2017, Guan et al. [7] proposed a delay-tolerant flexible data access control scheme based on key policy attribute-based encryption (ABE) for smart grids. Their scheme has no central trusted server to perform the encryption and decryption. But when the user revokes, the remote terminal unit (RTU) needs to redefine the access structure and recalculate part of the ciphertext, and this increases the overhead of RTU calculation and communication.

In 2008, Satoshi Naknamo proposed a new digital currency-Bitcoin [8], which is constructed on a Peer-to-Peer (P2P) network. Blockchain is the underlying core supporting technology of Bitcoin. Blockchain is mainly divided into three types: public blockchain, consortium blockchain and private blockchain. Public blockchain is a fully decentralized distributed architecture, registration and presentation of nodes is frequent and the number of nodes is constantly changing. Today's digital currencies, e.g. bitcoin, are traded on the public blockchain. Consortium blockchain is a relatively stable blockchain of nodes. The joining or exiting of each participating node in the consortium blockchain requires permission. Private blockchain generally has a fully trusted controller. To some extent, private blockchain has lost the meaning of decentralization. Consensus mechanism makes blockchain to be a decentralized distributed ledge system. Public blockchain generally uses a single proof mechanism to achieve consensus, such as Proof-of-Work (PoW), Proof-of-Stake (PoS). PoW is anonymized by Naknamo, whose main idea is to use computing power to find specific numbers to make the block to meet the requirements. PoW consensus algorithm solves the consensus problem in a completely decentralized network. At the same time, PoW brings defects of low system efficiency, waste of resources. Therefore, the scenario of applying Pow is very limited. In 2012, King and Nadal proposed the concept of PoS [9], it use the stake to replace or partially replace computing resources. PoS is more resource efficient than PoW, and the creation of blocks is no longer limited to hash calculations that satisfy high difficulty coefficients. However, PoS consensus algorithm is easy lead to uneven wealth, because rich nodes always have advantage to be chosen as ledgers. Apart from these, Larine proposed Delegate Proof-of-Stake (DPoS) which each shareholder has a certain voting right [10]. Compared to the public blockchain, consortium blockchain and private blockchain can achieve consensus without relying on computing resources, and only need to improve the underlying consensus agreement. The Byzantine Fault Tolerant Algorithm (PBFT) is a consistency algorithm based on state machine replication proposed by Castro and Liskov [11], and is widely used in distributed systems. In the environment of asynchronous communication, the algorithm can guarantee the safety and liveness [12] of the system under the failure node of no more than $\lfloor \frac{n-1}{3} \rfloor$. In a limited number of nodes, the efficiency of the PBFT is considerable. But if the number of nodes increases, the quality of the service provided will decrease. In addition, there are some other consensus algorithms, such as Raft [13] and Paxos [14].

In 2017, Maesa et al. explored how to formulate the classical access control scheme as a smart contract that can be stored and executed in the blockchain [15]. In 2018, Lin et al. proposed a novel blockchain-based framework to ensure a secure user authentication with fine-grained access control [16], which used Attribute-based signature (ABS). Both of them only consider signature or encryption, and did not prove the strict security proof in the random oracle model (ROM).

In 2011, the concept of a combined public key cryptosystem was first proposed by Haber and Pinkas [17]. They showed that reusing a single key pair during encryption and signature does not compromise the security of the solution. That is, in an signature scheme, an adversary can access the decryption oracle in an encryption scheme. In addition, in an encryption scheme, an adversary can access the signature oracle in a signature scheme, which does not pose any security threat to an encryption scheme. In 2015, Vasco et al. [18] constructed an identity-based combined public key cryptography scheme, and proved that the Hess identity-based signature (IBS) scheme [19] and the Boneh and Franlins identity-based encryption (IBE) scheme [20] can be safely combined. In 2017, Zhou and Li proposed an identity-based combined public key scheme for signature, encryption and signature (IBCSESC) [21]. Under the premise of ensuring the confidentiality, integrity, authentication and non-repudiation of data, the combined cryptosystem reduces the key management work, saves storage space and computational consumption, and is very suitable applied for complex grid environment.

B. Motivation and Contribution

At present, power grids generally usually store information in plaintexts. The existing access control schemes generally include a trusted third party, which is easy to suffer from key escrow attacks. The idea of decentralization of the blockchain can solve the problem of key escrow.

This paper focuses on the premise that there are many smart grid nodes and many application scenarios. We makes the following contributions:

1) We give a consensus algorithm for the selection of private key generator (PKG) in smart power grids. This consensus algorithm not only has an incentive mechanism, but also has a penalty mechanism, which is not currently available in consensus algorithms.

2) We design a blockchain-based access control scheme that uses a combined cryptosystem. Our scheme satisfies the security requirements of power grids, solves the key escrow problem, and makes users more involved in the daily management of smart grids. The reuse of keys saves

system communication costs, and is also very suitable for different services in smart grids multi-environments.

C. Organization

The rest of the paper is arranged as follows. The network model and security requirements are introduced in section II. The blockchain-based access control scheme is proposed in section III. The security analysis for our scheme is given in section IV and the performance analysis is given in section V. Finally, the conclusions are given in section VI.

II. Preliminaries

A. Network Model

The blockchain-based access control scheme proposed in this paper is showed in Figure 1. In Figure I, we mentioned that smart grids consist of three user types (i.g. HAN, BAN and IAN). In this paper, we take HAN as an example to describe our access control scheme. The network model includes three parts: a HAN, a power provider and a PKG.

1) HAN: Smart meter can collect the user's power consumption data, which will be monitored by the master controller. Master controller in the HAN is used to manage the user's power consumption and establish communication with the power provider.

2) Power provider: Power provider is an electric power company, which mainly produce, transmit and sell electricity. Power provider can customize the power consumption scheme according to the power consumption information returned by the user, and inform the user through the e-mail and so on. At the same time, in smart girds, some users can sell their collected excess power to the electric power company. Therefore, the flow of information in smart grids is two-way. Power provider uploads the information in smart grids to the cloud system.

3) PKG: A PKG will generate the user's and power provider's private key. In this paper, PKG is a consensus reached by multiple HAN user nodes and multiple servers in the power provider.

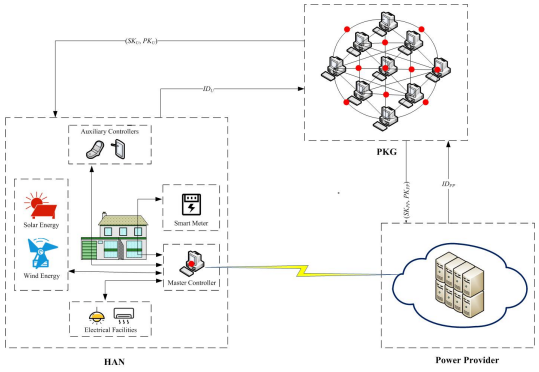

Fig. 1. Smart grids network model

B. Security Requirements

The secure communication of data in smart grids needs to meet the following security requirements.

1) Confidentiality: Privacy data can only be accessed by authorized users, i. e. the message should satisfy indistinguishability against adaptive chosen ciphertext attacks under chosen identity attacks (IND-ID-CCA2);

2) Integrity: Data will not be tampered by illegal attackers, i. e. the message should satisfy existential unforgeability against adaptive chosen messages attacks under chosen identity attacks (EUF-ID-CMA);

3) Identity authentication: The sending and receiving of data must be legitimate users. The authentication includes the legal authentication of the user and the legal authentication of the device;

4) Non-repudiation: Users cannot deny the data they sent or received.

III. A Blockchain-based Access Control Scheme for Smart Grids

Most access control schemes in smart grids require a fully trusted third-party, e.g. PKG. However, if PKG is replaced by a malicious adversary, then this malicious adversary can very easy to know users' and power providers' private keys. And this adversary can deceive both parties to get information they want (e.g. man-in-the-middle attack). Considering the decentralization of the blockchain, we decided to use the consensus mechanism to solve the third-party trust problem. The two-way communication in smart grids also enhances the interaction between users and power providers. Some scenarios only need to encrypt the message (e.g upload users' private information to the power cloud system), some scenarios only need to sign the message (e.g. individual users with sufficient power storage apply to sell excess electricity to the power provider), and some scenarios need signcrypt message (e.g. power provider apply to access users' private information). In order to reduce the storage space of keys and grid system's communication costs. Our scheme based on Boneh and Franklin's IBE scheme [20], Cha and Cheon's IBS scheme [25], Zhou and Li's IBCSESC scheme [21].

A. Consensus Mechanism

Nodes in the public blockchain can be accessed or exited at any time without permission. However, in smart grids, nodes are generally relatively fixed. Public blockchain generally uses a single proof mechanism to achieve consensus, but most of the proof mechanisms would waste a lot of resources, such as Pow, which needs to use workload. Therefore, we have chosen PKG to bulid on consortium blockchain, in which each participating node joining or withdrawing requires permission. Our scheme is based on the PoS consensus algorithm, but we specify that the node does not have dynamics like in PoS. Every node that wants to campaign for the PKG must be the relatively fixed user node and server node inside the power provider. One HAN represents a user node. These nodes must be registered full network, submit to a representative audit

system, such as on the local national department's website. It can be said that representative audit system is also a trusted third party, but this must be controlled by the state or government. At the same time, draw inspiration from FBFT algorithm, we also divide the nodes in HAN into master and slave nodes. Only the master node can participate in the campaign PKG. Slave only forward the received transaction data, and participate in confirming the generation of PKG. The master-slave of the node is confirmed by the auditing agency according to its social credit rating when node is registered. Social credit refers not only to the credit evaluation of users participating in smart grids, but also the behavioral credit evaluation of the user in social activities. Therefore, the auditing agency must be a national-level organization, otherwise it would not be able to collect information on social activities of ordinary users.

We treat every user node or server node as a block and the output block is the PKG. The consensus algorithm is showed below. Let every candidate node is $B =< nonce, txs, preHash >$. $nonce$ is a integer and changing any bit $nonce$ will completely change the hash value of the entire nodes. txs is transaction records contained in the block. $preHash$ is the hash value of the previous block. D is the difficult value and defines how many leading zeros are needed for the current hash value of the entire block. The more the leading zeros, the more difficult it is. In order to prevent users from consuming a lot of computing resources, this D can also automatically adjust the parameters so that D is a suitable value. The detailed description of the consensus algorithm is showed in algorithm 1 below.

Algorithm 1 Consensus Algorithm

Input: $preHash, txs, D, energyUsed, HashTransactionTime$;
Output: $block$

$nonce \leftarrow 1$
$coins \leftarrow energyUsed$
$age \leftarrow currentTime - hashTransactionTime$
while $(H(nonce, txs, preHash) \geq coins \cdot age \cdot D$ **do**
  $nonce \leftarrow nonce + 1$
  $Broadcast(< nonce, txs, preHash >)$
end while
if $(H(nonce, txs, preHash) < coins \cdot age \cdot D$ **then**
  The node that
  first $Broadcast(< nonce, txs, preHash >)$ is PKG
end if
return $block$

In a period of time, the more electricity users use, the easier it is to be chosen to be PKG. The server node inside the power provider will receive a certain percentage of $energyUsed$ based on the amount of electricity it delivers. Note that the slave node does not participate in the campaign for PKG. However, in the sixth step broadcast, it participates in confirming the generation of PKG. When a PKG campaign ends, age will be cleared. In order to allow more nodes to participate in the PKG campaign, the nodes that have elected PKG cannot be reelected within a certain period. This makes the election more fair.

In addition, our consensus algorithm also has incentive mechanism and penalty mechanism. If a node successfully wins to be PKG and runs safely during the period of its responsibility, there will be certain rewards, such as the free electricity. However, if a information disclosure accident happens, the node that is acting as PKG this time, its credit evaluation will be dropped significantly and can never to be a master node.

B. A Detailed Access Control Scheme

In this subsection, we design a based access control scheme for smart grids, in which we use the blockchain consensus algorithm to genarate PKG. PKG generate the users' and power providers' private key according to their ID. The scheme consists of four parts: initial phase, registration phase, verification and authorization phase, and withdrawal phase. Figure 2 summarizes these five phases.

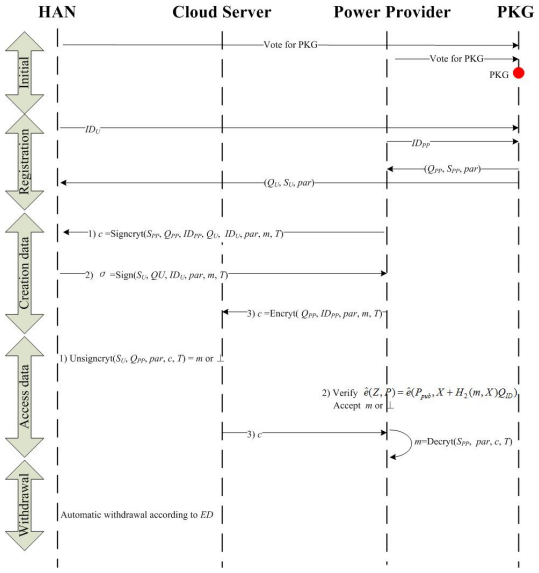

Fig. 2. Access control scheme

- ***Initial phase:*** In this phase, the nodes with voting rights in smart grids ran for PKG. Note that this PKG is only for a while. If user's private keys are compromised during this time, or if a serious malicious incident occurs, the node playing the PKG will be deprived of PKG voting rights and campaign rights.
  Given a security parameter k, the PKG selects an additive group G_1, a multiplicative group G_2, a bilinear pair $\hat{e}$, and five hash functions $H_1 : \{0,1\}^n \rightarrow G_1$, $H_2 : \{0,1\}^n \times G_1 \rightarrow \mathbb{Z}_q^*$, $H_3 : \{0,1\}^n \times \{0,1\}^n \rightarrow \mathbb{Z}_q^*$, $H_4 : G_2 \rightarrow \{0,1\}^*$, $H_5 : \{0,1\}^* \rightarrow \{0,1\}^*$, $H_6 : G_2 \rightarrow \{0,1\}^{|G_1|+|ID|+n}$. The group, whose order is prime q and the generator of G_1 is P. Bilinear pair is $\hat{e} : G_1 \times G_1 \rightarrow G_2$. n is the number bits of the encrypted or signed message, $|G_1|$ and $|ID|$ are the

number of bits in G_1 and ID. PKG selects a master key s and keeps it secret, then calculates own public key $P_{pub} = sP$. Finally, PKG public system public parameter $par = \{G_1, G_2, \hat{e}, n, P, P_{pub}, H_1, H_2, H_3, H_4, H_5, H_6\}$.

- ***Registration phase:*** In this phase, users in HANs and power providers obtain their public keys and private keys. User and power provider submit their own ID_U and ID_{PP} to PKG, and PKG check whether the ID is valid. If the ID is illegal, PKG rejects the applicant's registration request. Otherwise, PKG generates corresponding public key $Q_U = H_1(ID_U \parallel ED_U)$ and $Q_{PP} = H_1(ID_{PP} \parallel ED_{PP})$, private keys $S_U = sQ_U$ and $S_{PP} = sQ_{PP}$ according to their ID. Note that, ED_U and ED_{PP} are the an access validity periods for user and power provider applicants. PKG sends (S_U, S_{PP}) to user and power provider online or offline. If online transmission is used, then we can use the secure socket layer (SSL) protocol to ensure the confidentiality of private key.
- ***Data creation phase:*** In this phase, we have three scenarios.

  1) We assume that a power provider with identity ID_{PP} want to access the data of a user with identity ID_U. The power provider first generate a query message m. In order to resist the replay attack, the power provider concatenates m and a timestamp T to form a new message that is $m^* = m \parallel T$. The power provider random selects $r \in \mathbb{Z}_q^*$, then calculates $X = rQ_{PP}$, $h = H_2(m^*, X)$, $Z = (r + h)S_{PP}$, $\omega = \hat{e}(rS_{PP}, Q_U)$, and $y = H_6(\omega) \oplus (Z \parallel ID_{PP} \parallel m^*)$. The ciphertext is $c = (X, y)$. Finally, power provider sends c to user.

  2) We assume a user with identity ID_U want to sell his excess electricity to a power provider. Users only need to verify their legal identity ID_U in smart grids to the power provider. User first generate a query message m. In order to resist the replay attack, the user concatenates m and a timestamp T to form a new message that is $m^* = m \parallel T$. Then in order to gain anonymity, the user calculates $X = rQ_U$. Next, user calculates $h = H_2(m^*, X)$ and $Z = (r+h)S_U$. The signature is $\sigma = (X, Z)$. Finally, user sends σ to power provider.

  3) We assume that power provider with identity ID_{PP} want to save user information to cloud storage servers. So he only needs to encrypt the message m with his public key Q_{PP}. When he want to acquire m, he decrypt ciphertext with his private key S_{PP}. In order to resist the replay attack, power provider generate a new message $m^* = m \parallel T$. First, power provider random selects $\lambda \in \{0,1\}^n$ and computes $t = H_3(\lambda, m)$. Next, power provider compute $U = tP$, $V = \lambda \oplus H_4(\hat{e}(Q_{PP}, P_{pub})^t)$ and $W = m \oplus H_5(\lambda)$. The cipertext is $c = (U, V, W)$. Power provider upload c to the cloud storage server.

- ***Data access phase:*** In this phase, we have different access data phases for the above three scenarios.

  1) When receiving the ciphertext c, user first compute $\omega = \hat{e}(X, S_{ID_U})$, $Z \parallel ID_{PP} \parallel m^* = y \oplus H_6(\omega)$ and $h = H_2(m^*, X)$. Next, user verifies $\hat{e}(Z, P) = \hat{e}(P_{pub}, X + hQ_{PP})$ is true. If it holds, user accepts $m = m^* \parallel T$. Otherwise, user rejects c and outputs $\perp$.

  2) In order to verify whether σ is the valid signature under message m sent by the user with identity ID_U, power provider should first compute $h = H_2(m^*, X)$. Then, if $\hat{e}(Z, P) = \hat{e}(P_{pub}, X + hQ_U)$ holds, power provider accept $m = T \parallel m^*$. Otherwise, power provider rejects σ and outputs $\perp$.

  3) When the power provider downloads the ciphertext c from the cloud server, he uses his private key S_{PP} to decrypt c. First, power provider computes $\lambda = V \oplus H_4(\hat{e}(U, S_{PP}))$. Next, power provider computes $m = W \oplus H_5(\lambda)$, $t = H_3(\lambda, m)$, and verifies $U = tP$ is true. If it holds, power provider will accept m, otherwise rejects c. This ensures that the message m will not be stolen on the cloud storage server.

- ***Withdrawal phase:*** In this phase, users in HAN and power providers registration are automatically revoked due to expiration of the expiration date ED. For example, if the due date is "2019-12-28", then their private keys are valid before December 28,2019. If for some special reason, the deadline is advanced. Then PKG will broadcast the identity of the revoked user identity and create a table to hold the identity of these invalid users. At the same time, we can also use Tsai and Tseng's methods [22] to revoke the power of user access.

In the above ***Registration phase*** and ***Data access phase***, we can see the advantage of our access control scheme. That is, the same key can be reused under different requirement of scenarios. Because of the complexity of the scenarios, access control in smart grids often cannot be covered by a single scheme. However, in our access control scheme, users in smart girds can use only one key to encrypt, signature and signcryt message. No matter users in what scenario, only one key can meet the user's access control needs. Therefor, our scheme simplifies the access control program, saves the storage space of keys, reduce the communication cost of the system and also guarantee the confidentiality, integrity, and non-repudiation of the verification message.

IV. Security Analysis

In this section, we analyze the security of our proposed access control scheme. In the subsection II-B, we defined a access control scheme in smart grids need to meet the security requirements: confidentiality,integrity, identity authentication, non-repudiation.

Because our scheme uses signature with user's identity, so we can achieve identity authentication. At the time, our scheme use hash function for message, so a message creator cannot

deny the fact that he made the message, that means our scheme achieve non-repudiation. Our access control scheme is based on Boneh and Franklin's IBE scheme [20], Cha and Cheon's IBS scheme [25].So under that these two schemes have been proven to be safe, we prove that our access control scheme satisfies the confidentiality and integrity by following 1 and 2, respectively.

Theorem 1: Our proposed access control scheme satisfies IND-ID-CCA2 security in the random oracle model.

Proof: Our proposed access control scheme uses same key to achieve encryption and signcryption, so we should prove the IND-ID-CCA2 security of the encryption part and the IND-ID-CCA2 security of the signcryption part.The IND-ID-CCA2 security of our proposed access control scheme is defined through the following game played between a challenger $\mathcal{C}$ and an adversary $\mathcal{A}$.

1) If an adversary $\mathcal{A}$ can use the q_k key extraction queries, q_d decryption queries, q_s signature queries to break the $(\epsilon, t, q_k, q_d, q_s)$-IND-ID-CCA2 security of the encryption part of our proposed access control scheme with a non-negligible advantage ϵ and in a bounded time t. We can construct a challenger $\mathcal{C}$ that can break the Boneh and Franklin's IBE scheme's (ϵ, t, q_k, q_d)-IND-ID-CCA2 security.

- ***Initial:***We assume that the system public parameters of Boneh and Franklin's IBE scheme are $par_{bf} = (G_1, G_2, n, \hat{e}, P, P_{pub}, H_1, H_3, H_4, H_5)$, and $\mathcal{C}$ choose H_2 and sends $par = (G_1, G_2, n, \hat{e}, P, P_{pub}, H_1, H_2, H_3, H_4, H_5)$ to $\mathcal{A}$.
- ***Phase 1:***
  a) When $\mathcal{A}$ asks a key extraction query with an identity ID, $\mathcal{C}$ submits the ID to its key extraction oracle and returns the result to $\mathcal{A}$.
  b) When $\mathcal{A}$ asks a decryption query with a (c, ID), $\mathcal{C}$ submits the (c, ID) to its decryption oracle and returns the result to $\mathcal{A}$.
  c) When $\mathcal{A}$ asks a signature query with a (m, ID), $\mathcal{C}$ randomly selects $r \in \mathbb{Z}_q^*$,computes $h = H_2(m, X)$,$X = rP - hQ_{ID}$, $Z = rP_{pub}$ and returns the result $\sigma = (X, Z)$ to $\mathcal{A}$.
- ***Challenge:***$\mathcal{A}$ generates two plaintexts m_0 and m_1 of the same length, one user identity ID^* that is intended to challenge, and $\mathcal{A}$ cannot request the private key of ID^* in ***Phase 1***. C sends m_0, m_1, ID^* to oracle and gets the ciphertext $c^* = Encrypt_{ID^*}(m_\gamma)$, then returns c^* to $\mathcal{A}$.
- ***Phase 2:***
  a) $\mathcal{A}$ can perform a polynomial bounded number of queries as in ***Phase 1***.
  b) $\mathcal{A}$ cannot make a key extraction query for ID^*.
  c) $\mathcal{A}$ cannot make a decryption query for (c^*, ID^*).
- ***Guess:*** $\mathcal{A}$ outputs a bit γ' and wins the game if $\gamma' = \gamma$.

2) If an adversary $\mathcal{A}$ can use the q_k key extraction queries, q'_d decryption queries, q'_s signature queries, q'_{sc} signcryption queries, q'_{usc} unsigncryption queries to break the $(\epsilon, t', q_k, q'_d, q'_s, q'_{sc}, q'_{usc})$-IND-ID-CCA2 security of the signcrtption part of our proposed access control scheme with a non-negligible advantage ϵ and in a bounded time t'. We can construct a challenger $\mathcal{C}$ that can break the Boneh and Franklin's IBE scheme's (ϵ, t, q_k, q_d)-IND-ID-CCA2 security, in which $q_d = q'_d + q'_{usc}$, $t = t' + O(q'_{sc}T_e + q'_{usc}T_v)$, T_e and T_v are the maximum time spent calculating an encryption and verifying a signature.

- ***Initial:***We assume that the system public parameters of Boneh and Franklin's IBE scheme are par_{bf}. $\mathcal{C}$ choose H_2, H_6 and sends $par = (G_1, G_2, n, \hat{e}, P, P_{pub}, H_1, H_2, H_3, H_4, H_5, H_6)$ to $\mathcal{A}$.
- ***Phase 1:***Apart from the following two queries, $\mathcal{C}$ can make the same answers as the ***Phase 1*** in the previous game according to $\mathcal{A}$'s queries.
  a) When $\mathcal{A}$ asks a signcryption query with a (m, ID_{PP}, ID_U), $\mathcal{C}$ randomly selects $r \in \mathbb{Z}_q^*$, computes $X = rP - hQ_{ID_{PP}}$, $h = H_2(m, X)$, $Z = rP_{pub}$, $\omega = \hat{e}(r - hQ_{ID_{PP}}/P, Q_{ID_U}P_{pub})$, $y = H_6(\omega) \oplus (Z \parallel ID_{PP} \parallel m)$ and returns the result $\sigma = (X, Z, y)$ to $\mathcal{A}$.
  b) When $\mathcal{A}$ asks an unsigncryption query with a (c, ID_{PP}, ID_U), $\mathcal{C}$ creates a table L_2 to store $\mathcal{A}$'s queries and H_2's answers, and checks whether the record (ω, m, r) in L_2 satisfies $X = rQ_{PP}$, $h = H_2(m, X)$,and $y = H_6(\omega) \oplus (Z \parallel ID_{PP} \parallel m)$. If it is included, the unsigncryption result is m, otherwise it returns $\perp$.
- ***Challenge:***$\mathcal{A}$ generates two plaintexts m_0 and m_1 of the same length, one sender identity ID^*_{PP} and one receiver identity ID^*_U that is intended to challenge, and $\mathcal{A}$ cannot request the private key of ID^*_{PP} in ***Phase 1***. C sends m_0, m_1, ID^*_{PP}, ID^*_U to the oracle and gets the corresponding ciphertext $c^* = Signcrypt_{ID^*_{PP}, ID^*_U}(m_\gamma)$, then returns c^* to $\mathcal{A}$.
- ***Phase 2:***
  a) $\mathcal{A}$ can perform a polynomial bounded number of queries as in ***Phase 1***.
  b) $\mathcal{A}$ cannot make a key extraction query for ID^*_U.
  c) $\mathcal{A}$ cannot make a unsigncryption query for (c^*, ID^*_{PP}, ID^*_U), which means $\mathcal{C}$ cannot submit (c^*, ID^*_U) to its decryption oracle.
- ***Guess:*** $\mathcal{A}$ outputs a bit γ' and wins the game if $\gamma' = \gamma$.

■

Theorem 2: Our proposed access control scheme satisfies EUF-ID-CMA security in the random oracle model.

Proof: Our proposed access control scheme uses same key to achieve encryption and signcryption, so we prove the EUF-ID-CMA security of the signature part and the EUF-ID-CMA security of the signcryption part.The EUF-ID-CMA security of our proposed access control scheme is defined through the following game played between a challenger $\mathcal{C}$ and an forgery $\mathcal{F}$.

1)If an adversary $\mathcal{F}$ can use the q_k key extraction queries, q_d decryption queries, q_s signature queries to break the $(\epsilon, t, q_k, q_d, q_s)$-EUF-ID-CMA security of the signature part of our proposed access control scheme with a non-negligible

advantage ϵ and in a bounded time t. We can construct a challenger $\mathcal{C}$ that can break the Cha and Cheon's IBS scheme (ϵ, t, q_k, q_s)-EUF-ID-CMA security.

2)If an adversary $\mathcal{F}$ can use the q_k key extraction queries, q'_d decryption queries, q'_s signature queries, q'_{sc} signcryption queries, q'_{usc} unsigncryption queries to break the $(\epsilon, t', q_k, q'_d, q'_s, q'_{sc}, q'_{usc})$-EUF-ID-CMA security of the signcrtption part of our proposed access control scheme with a non-negligible advantage ϵ and in a bounded time t'. We can construct a challenger $\mathcal{C}$ that can break the Cha and Cheon's IBS scheme (ϵ, t, q_k, q_s)-EUF-ID-CMA security, in which $q_s = q'_s + q'_{sc}$, $t = t' + O(q'_{sc}T_e + q'_{usc}T_v)$, T_e and T_v are the maximum time spent calculating an encryption and verifying a signature.

Except the attacker's queries parts, the content for EUF-ID-CMA security of our scheme is similar to the proof of EUF-ID-CMA security in [21], which we will not prove again in this paper. If you have any questions, please contact us by email.

■

In addition, unlike the incentive mechanism of Bitcoin, our scheme can take penalty mechanism, which means that the designated mode will be blacklisted and isolated once he refuses to ensure the safe operation of the power system. We assume that more than half of the nodes in the system network are honest, so we can conclude that our access control scheme satisfied the security requirements in the subsection II-B.

V. Performance Analysis

In this section, we discuss our access control scheme's performance. After reviewing papers, we compared it with Paerson et al. [23], Vasco et al. [18] and Wang et al. [24]. Table I shows the components of each scheme. Symbol "$\surd$" indicates the scheme has this feature and symbol "$\times$" indicates the scheme has not this feature. Table I shows our comparison results.

TABLE I
COMPARISON OF SCHEMES COMPONENT

Scheme	Encryption	Signature	Signcryption	Crypotosystem
[23]	$\surd$	$\surd$	$\times$	IBC
[18]	$\surd$	$\surd$	$\times$	IBC
[24]	$\times$	$\times$	$\surd$	ABC
ours	$\surd$	$\surd$	$\surd$	IBC

Because the addition operation, exponent operation, pairing operation, and point multiplication operation are the most expensive in the whole scheme, and other operations are negligible compared with them. So we use these four operation as a measure to calculate the basic operation of cost. Table II shows the comparison of the calculation costs and communication costs of these schemes. Here, we use PM to denote the point multiplication operation in the group G_1, use Exp to denote the exponent operation in the group G_2, and use P to denote the pairing operation on the bilinear map. For communication costs, we use $|m|$ to denote the number of bits of message m, $|G_1|$ indicates the number of bits of an element in group G_1, $|G_2|$ indicates the number of bits of an element in group G_2, $\left|\mathbb{Z}_q^*\right|$ indicates the number of bits of an element in the group $\mathbb{Z}_q^*$, $|ID|$ indicates the number of bits of the group ID . At the same time, because of the attributed-based scheme, we use l to indicate the length of the attribute set involved, n_s to represent the number of group members in the self-organizing network, and $|S|$ to represent the number of bits in the attribute organization.

TABLE II
COMPARISON OF SCHEMES PERFORMANCE

Scheme	Computation cost		Communication cost
	Sender	Receiver	
[23]	5PM+4Exp	2Add+3PM+Exp+3P	$3\|G_1\| + \left\|\mathbb{Z}_q^*\right\| + \|G_2\|$
[18]	Add+3PM+2Exp+2P	PM+Exp+3P	$3\|m\|+3\|G_1\|$
[24]	$(3l + n_s + 3)$Exp	$(l + n_s)$Exp+(3+2l)P	$(2l+3)\|G_1\|+\|G_2\|+\|S\|$
ours	3PM+Exp+P	Add+PM+3P	$2\|G_1\|+\|m\|+\|ID\|$

From Table II, we can see that our scheme has fewer number of point multiplication operation, exponential operations, and pairing operations than [23] and [18]. Attribute-based cryptosystem universal storage makes the private key length too long. It also has the characteristics of the computation cost, the length of the attribute set and the number of group members in the self-organizing network are linearly increased. Therefore, the calculation cost in the scheme [24] cannot obtain a value in this paper. It can only be known that the scheme [24] does not have any advantage in terms of computation cost and communication cost.

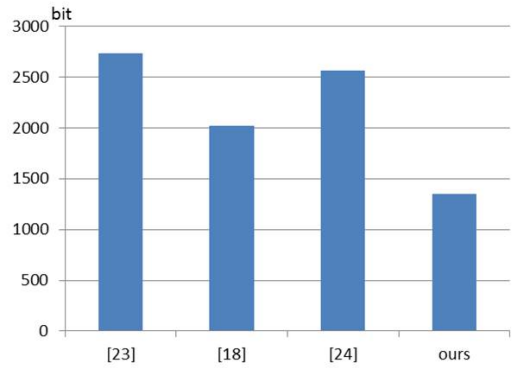

Fig. 3. Comparison in communication cost

In order to more intuitively draw the advantages of our scheme in communication cost, we set $|m|$=160 bits, $|ID|$=160 bits, $|G_1|$ = 513 bits, $|G_2|$ = 1024 bits, $\left|\mathbb{Z}_q^*\right|$= 169 bits. Figure 3 shows the comparison of communication costs of these schemes. For scheme [24], we set the length of the attribute set l in 0 bit, and the number of bits $|S|$ of the elements in the attribute mechanism is 0 bit. In reality, l and $|S|$ cannot be 0 bits. Our scheme also dominates the communication

cost. As can be seen from Table II and Figure 3, our scheme has certain advantages in computing cost and communication cost in theoretical analysis. In particular, our scheme is also the only one of the four schemes that implements combined encryption, signature, and signcryption. Next, we implemented the access control scheme using the JPBC library.

Constructing a bilinear pair here, we use a symmetric pairing based on the elliptic curve $y^2 = x^3 + x$ mod p in the finite field $E(\mathbb{F}_p)$. Considering the security of the protocol, we take p=512 bits, the order q of the cyclic group is a large prime number of 160 bits. So the output of H_2 and H_3 is 160 bits. Since G_1 is a cyclic addition group on the finite field $E(\mathbb{F}_p)$, P is the generator of G_1, so the size of P is 1024 bits. The size of P_{pub} is 1024 bits, and the output of the Hash function H_1 is also 1024 bits. Here we use the secure Hash function SHA-256, so the H_4, and H_5 output is 256 bits.

We implement our scheme is Eclipse, Neon.1a Release (4.6.1). The computer configuration of the program execution environment is: Intel(R) Core(TM) i5-5200U CPU @ 2.20GHz 2.19GHz processor, 8GB of RAM, 64-bit Windows operating system. In order to make the experimental values more representative, we cycle through the entire steps of the access control scheme 1000 times to get the average time taken to complete each algorithm. In scenario 1, we only need to encrypt and decrypt message to achieve our access control, so the the initial phase time is 105 ms, the creation phase time is 24 ms, the access data phase is 21 ms. In scenario 2, we only need to sign and verify message to achieve our access control, so the the initial phase time is 105 ms, the creation phase time is 73 ms, the access data phase is 76 ms. In scenario 3, we only need to signcrypt and unsigncrypt message to achieve our access control, so the the initial phase time is 105 ms, the creation phase time is 95 ms, the access data phase is 81 ms. From the figure 4, we can know that the initial phase has a large proportion of time in every scenario. If the user needs to implement all three scenarios, the proportion of time in the initial phase will be greatly increased, then the computational cost of the system will increase greatly. Conversely, if the user uses our scheme, the initial phase time will be reduced to 1/3 of the original.

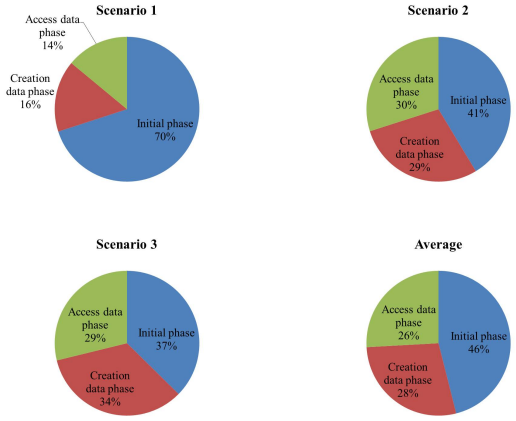

Fig. 4. Ratio of each phase running time

VI. Conclusions

In this paper, we proposed an access control scheme for smart grids based on blockchain technology. In our scheme, we use a consensus mechanism based on the consortium blockchain, which solves the trust problem of PKG. At the same time, we use a combined cryptosystem to enable our access control scheme to cope with as many scenarios as possible. Analysis shows that the proposed scheme has lower communication cost compared with the scheme of the same type. Therefore our proposed access control scheme is very suitable for application in practical smart grids.

References

[1] X. Fang, S. Misra, G. Xue, D. Yang: Smart Grid-The New and Improved Power Grid: A survey. In IEEE Communications Surveys & Tutorials, vol. 14, no. 4, pp. 944–980 (2011)

[2] A. Mahmood, N. Javaid, S. Razzaq: A Review of Wireless Communications for Smart Grid. In Renewable and Sustainable Energy Reviews, vol. 41, pp. 248–260 (2015)

[3] Y. Yuang, F.Y. Wang: Blockchain: The State of the Art and Future Trends. In Acta Auto-matica Sinica, vol. 42, no. 4, pp. 481–494 (2016)

[4] L. Sankar, S. Kar, R. Tandon, et al.: Competitive Privacy in the Smart Grid: An Infor-mation-theoretic Approach. In Pro. IEEE International Conference on Smart Grid Com-munication, pp. 220–225 (2011)

[5] Z.W. Sun, R.G. Zhang: Access Control for communication Network of Smart Distribution Grid. In Power System Protection and Control, vol. 21, no. 38, pp. 118–121 (2010)

[6] J. Wu, M.X. Dong, et al.: Toward Fault-Tolerant Fine-Grained Data Access Control for Smart Grid. In Wireless Personal Communications, vol. 75, no. 3, pp. 1787-1808 (2014)

[7] Z.T. Guan, J. Li, et al.: Toward Delay-Tolerant Flexible Data Access Control for Smart Grid With Renewable Energy Resources. In IEEE Transactions on Industrial Informatics, vol. 13, no. 6, pp. 3216–3225 (2017)

[8] S. Nakamoto: Bitcoin: a-peer-to-peer electronic cash system. In https://www.coindesk.com/bitcoin-peer-to-peer-electronic-cash-system, (2008)

[9] S. King, S. Nadal: Ppcoin: Peer-to-peer crypto-currency with proof-of-stake. Self-published paper, (2012)

[10] D. Larine: Delegated Proof-of-Stake (DPoS). In http://docs.bitshares.org/bitshares/dpos.html, (2014)

[11] M. Castro, B. Liskov: Practical byzantine fault tolerance and proactive recovery. In ACM Transactions on Computer Systems, vol. 20, no. 4, pp. 398–461 (2002)

[12] L. Lamport: Proving the Correctness of Multiprocess Programs. In IEEE Transactions on Software Engineering, vol. SE-3, no. 2, pp. 125–143 (1977)

[13] D. Ongaro, J. Ousterhout: In search of an understandable consensus algorithm. In Pro. 2014 USENIX Annual Technical Conference, pp. 305–319 (2014)

[14] L. Lamport: Generalized Consensus and Paxos. In Microsoft Research, vol. 7, no. 7, pp. 809–812 (2005)

[15] D. D. F. Maesa, P. Mori, L. Ricci:Blockchain Based Access Control. In Pro. 17th IFIP WG 6.1 International Conference, pp. 206C-220 (2017)

[16] C. Lin, D. He, et al: BSeIn: A blockchain-based secure mutual authentication with fine-grained access control system for industry 4.0. In Journal of Network and Computer Ap-plications, vol. 116 ,pp. 42–52 (2018)

[17] S. Haber, B. Pinkas: Security combining public-key cryptosystem. In Pro. 8th ACM on CCS 2001, pp. 215–224 (2001)

[18] M.I.G. Vasco, F. Hess, R. Steinwandt: Combined schemes for signature and encryption: The public-key and identity-based setting. In Information and Computation, vol. 247, pp. 1–10 (2016)

[19] F. Hess: Efficient identity based signature schemes based pairing. In Spring-Verlag Berlin Heidelberg, pp. 310–324 (2002)

[20] D. Boneh, M. Franklin: Identity-based encryption from the weil pairing. In SIAM Journal of computing, vol. 32, no. 3, pp. 586–615 (2003)

[21] Y.Y Zhou, Z.Q. Li, G. Hu, F.G. Li: Identity-Based Combined Public Key Schemes for Signature, Encryption, and Signcryption. In Pro. Information Technology and Applied Mathematics international conference 2017 , vol. 699, pp. 3–22 (2018)
[22] T.T. Tsai and Y.M. Tseng: Revocable certificateless public key encryption. In IEEE System Journal, vol. 9, no. 3, pp. 824–833 (2015)
[23] K.G. Paterson, J.C.N Schuldt, M. Stam, S. Thomson: On the joint security of encryption and signature, revisited. In Pro. International Conference on the Theory and Application of Cryptology and Information Security, pp. 161–178 (2011)
[24] C. Wang, X. Xu, Y. Li, et al: Integrating ciphertext-policy attributed-based encryption with identity-based ring signature to enhance security and privacy in wireless body area networks. In Information Security and Crytology, LNCS 8957, pp. 424-442 (2015)
[25] J.C. Cha, J.H. Cheon: An identity-based signature from gap Diffie-Hellman groups. LNCS, vol. 2567, pp. 18–30 (2003)