Journal of Electronics and Information Science (2023)
Clausius Scientific Press, Canada

DOI: 10.23977/jeis.2023.080105
ISSN 2371-9524 Vol. 8 Num. 1

V2G Blind Signature Privacy Protection Scheme Based on Blockchain

Baoyi Wang[a,*], Yakun Gou[b], Shaomin Zhang[c]

School of Control and Computer Engineering, North China Electric Power University, Baoding, Hebei, 071003, China
[a]wangbaoyiqj@126.com, [b]gouyakun@126.com, [c]zhangshaomin@126.com
[*]Corresponding author

Keywords: Electric vehicle, blind signature, privacy protection, bloom filter, blockchain

Abstract: Due to the open nature of the V2G network, it is extremely easy to expose the user's personal information. At present, there are many solutions to solve the problem of V2G privacy protection. However, most of these solutions cannot provide a fully distributed environment, and there is a hidden danger of single point of failure. This work adopts a V2G privacy protection system based on blind signing and distributed Bloom filter to address the aforementioned issues. The blind signature is used to ensure the anonymity and the unlinkability of pseudonyms. The Bloom filter is established in the blockchain to verify pseudonyms in a decentralized manner, thus improving the verification speed.

1. Introduction

Due to the development and promotion of V2G network, the number of Electric Vehicle (EV) users is increasing rapidly. The rapid development of V2G has changed people's lives. Yet there are also many difficulties with security and privacy issues in V2G networks, which are mostly seen in two areas: identity authentication and privacy protection services. They should verify one another before the Service Provider (SP) offers services to the EV. A malicious attacker may pose as an unlawful Charging Station (CS) in order to steal the identification and location details of EVs if mutual authentication is not enabled. In the current research, there have been a variety of methods used for V2G identity authentication and privacy protection [1]-[4]. For example, elliptic curve digital signature algorithm and one-way hash function are used for identity authentication, fog computing is used for billing identity authentication, revocation group signature is used for anonymous identity authentication of plug-in electric vehicles.

In order to achieve conditional anonymity, Wei et al. [5] presented a group blind signature system for smart grid. Homomorphic Encryption is used in this technique to verify the accuracy of consumption data, which lowers communication costs between the control center and the smart meter. The well-known Modified Digital Signature Algorithm was used by Zhang et al. [6] to construct an improved credential-based power request system architecture and to propose an untraceable blind signature method (MDSA). The structure was combined with the proposed blind signature. Blind signature technology has not been widely used in V2G field, but its high security

and low overhead make it very suitable for V2G environment. A privacy-protecting and efficient data aggregation mechanism was proposed by Guan et al. [7] for a smart grid that divided users into different groups and gave each subgroup a private blockchain to retain the information of its members. Users' identities were concealed within the group using pseudonyms, and each user was allowed to create many pseudonyms and link their data to various pseudonyms. Bloom filter is also utilized for quick authentication. The Key Generation Center (KMC) generates pseudonyms and the private keys that go with them in this manner, which creates a linkability issue because KMC may determine the user's identity from the pseudonym. In order to solve the shortcomings of previous research, A decentralized, entirely distributed ID system with privacy protection was created by Shin et al. [8]. The method provides unlinkability using blind signatures to ensure user anonymity. In addition to ensuring fully dispersed operation through the certificate authority in charge of user key management, it also offers the updating or revocation of pseudonyms using counting Bloom filters or revoking pseudonym Bloom filters.

In conclusion, the key points of this essay can be summed up as follows. First, blockchain is used for distributed storage in order to offer a completely distributed environment and handle the problem of possible single points of failure and centralized assaults. Users of electric vehicles and service providers also generate keys in order to provide a fully distributed environment. Secondly, aiming at the linkability problem between pseudonyms in the existing scheme, an improved blind signature algorithm was used to make pseudonyms unlinkable and unforgeable. Finally, Bloom filter was developed to verify pseudonyms in order to solve the poor effectiveness of pseudonym verification.

The rest of this article is divided into the following sections: part II covers similar techniques, such as ElGamal blind signature methods, blockchain methods, and Bloom filter methods. Part III introduces the paper's scheme model. The detailed implementation method of the system is described in the fourth chapter. The final portion serves as the paper's conclusion.

2. Related technology

2.1. ElGamal blind signature

The blind signature protocol is run between the EV user and the charging pile to obtain the specified blind signature. The EV signs a message but does not reveal the contents of the message.

The EIGamal[9] elements (p, g, x, y) should be considered, where p is a prime quantity picked at random, g being a multiplicative group Z_p^* generator, x being a randomly chosen parameter between [1, p - 2], and $x \in Z \cap [1, p - 2]$, $y = g^x \bmod p$. The corresponding private key of the user is x, and the user's public key is an array of values that looks like (y, g, p).

1) User A chooses the values of $\bar{k}$ and $\bar{k} \in$ [1, p-2], computes $\bar{r} = g^{\bar{k}} \, mod \, p$ and transmits $\bar{r}$ to user B.

2) User B selects two integers at random, $\beta_1, \beta_2 \in [1, p - 2]$ and computes two values r,$\bar{m}$,

$$r = m g^{\beta_1} \bar{r}^{\beta_2} \bmod p \tag{1}$$

$$\bar{m} = r\beta_2^{-1} \bmod (p - 1) \tag{2}$$

User B conveys user A the value $\bar{m}$.

3) User A computes and delivers to user B the value $\bar{s} = \bar{m}x + \bar{k} \bmod (p - 1)$.

4) User B figures out $s = \bar{s}\beta_2 + \beta_1 \bmod (p - 1)$.

A message m's blind signature tuple is (r, s).

2.2. Blockchain

Blockchain technology has been hailed as a breakthrough innovation due to its favorable characteristics and performance in a distrustful environment. This technology for a decentralized distributed ledger contains a number of time-stamped records. Every node holds a copy of the digital ledger on its own, therefore there is no need to rely on an independent third party. Every node in the network has access to and can independently verify every record kept on the blockchain. After the consensus mechanism successfully validates the transaction, a block is formed and added in chronological order. Each block is connected to the previous block by a distinct cryptographic hash, rendering it effectively unchangeable.

2.3. Bloom Filter

Using an efficient probability space data structure called a bloom filter, a set is effectively represented by encoding its components. False positives (FP) are permitted while rapid membership queries are supported by Bloom filters, and the Bloom filter parameters can be changed to minimize FP.

The algorithm is as follows:

1) First, we need k hash functions, each of which can hash the key to an integer.

2) To initialize, we need an array of n bits, each initialized to 0.

3) When a key is added to the set, the appropriate bit position in the array is assigned a value of 1, and k hash values are calculated using k hash functions.

4) Create k hash values using k hashing algorithm, then query the corresponding bits in the array to see if a key is present in the set. It is regarded as being in the set if all the bits are 1.

3. Scheme design

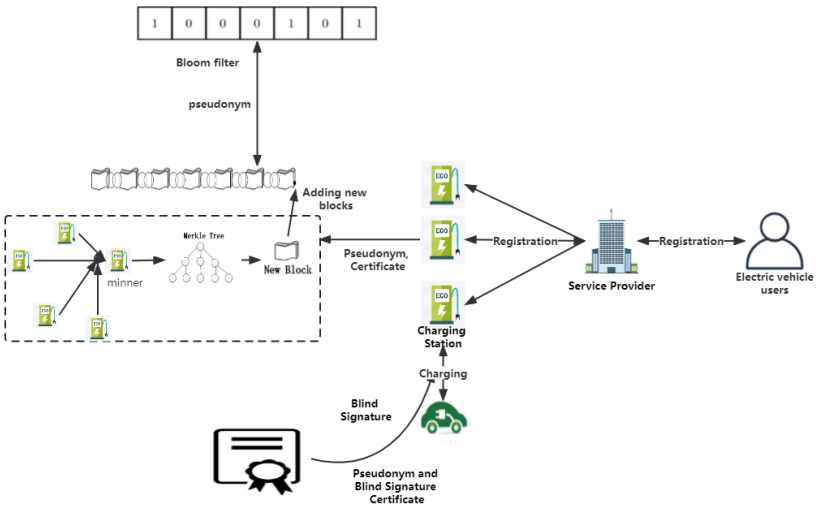

Figure 1: System architecture diagram.

A Based on the privacy protection framework of smart grid in reference [8], this paper designs a network architecture composed of Electric Vehicle (EV), Service Provider (SP), Charge Station (CS) and blockchain network. In Figure 1, this is displayed. The data center and power generation and distribution make up the service provider. Users register themselves with the service provider, and the service provider stores them in the data center, encrypts them through hash values, etc., and

generates pseudonymity to send to the electric vehicle. The service provider obtains electricity from different grid providers and allocates it to charging stations for the purpose of providing charging services for EVs.

4. Scheme implementation

4.1. Initialization

①These are the SP generation parameters: As the creator of the multiplicative group Z_p^*, SP produces a prime number P at random, chooses a random integer α, and finally selects a common hash function h: {0, 1}* → Z_p. Calculate $y_{sp} = \alpha^{x_{sp}} \bmod p$ after randomly generating the private key x_{sp}, $x_{sp} \in [1, p - 2]$. SP emits p, α, h and y_{sp}.

②The parameters of EV_i are generated as follows: the private key $x_{ev} \in [1, p - 2]$ is generated at random, $y_{ev} = \alpha^{x_{ev}} \bmod p$ is calculated, and y_{ev} is published.

③CS produces its parameters by calculating $y_{cs} = \alpha^{x_{cs}} \bmod p$, publishing y_{cs}, where the private key x_{cs} is randomly generated by CS and $x_{cs} \in [1, p - 2]$.

4.2. Registration

1) EV_i and SP authentication

①EV_i selects a random number $k_{ev_1} \in [1, p - 2]$, calculates $M_1 = \alpha^{k_{ev_1}} \bmod p$, $M_2 = m y_{sp}^{k_{ev_1}} \bmod p$, where m is EV_i's transmission message, EVi sends M_1,M_2 to SP;

②SP decrypts (M_1, M_2) to obtain m, $m = M_1^{-x_{sp}} M_2 \bmod p$, SP generates a random number $k_{sp_1} \in [1, p - 1]$ and $\gcd(k_{sp_1}, p - 1) = 1$, and then calculate $u_{ev} = \alpha^{k_{sp_1}} \bmod p$, $v_{ev} = k_{sp_1}^{-1}(h(m) - x_{sp}u_{ev}) \bmod (p - 1)$. SP stores ($u_{ev}$, v_{ev}), M_1, M_2, m in its data center and then sends (u_{ev}, v_{ev}) to EV_i;

③If $y_{sp}^{u_{ev}} u_{ev}^{v_{ev}} = \alpha^{h(m)} \bmod p$, EV_i accepts the signature (u_{ev}, v_{ev}), otherwise it rejects.

2) CS and SP authentication

①CS selects a random number $k_{cs} \in [1, p - 2]$, computes $M_3 = \alpha^{k_{cs}} \bmod p$, $M_4 = ID_{cs} y_{sp}^{k_{cs}} \bmod p$, where ID_{cs} is the identification ID of CS, CS sends M_3,M_4 to SP;

②SP decrypts (M_3, M_4) to obtain ID_{cs}, $ID_{cs} = M_3^{-x_{sp}} M_4 \bmod p$, SP generates a random number $k_{sp_2} \in [1, p - 1]$ and $\gcd(k_{sp_2}, p - 1) = 1$, Then calculate $u_{cs} = \alpha^{k_{sp_2}} \bmod p$, $v_{cs} = k_{sp_2}^{-1}(h(ID_{cs}) - x_{sp}u_{cs}) \bmod (p - 1)$. SP stores ($u_{cs}$, v_{cs}), M_3, M_4, ID_{cs} in its data center and then sends (u_{cs}, v_{cs}) to CS;

③If $y_{sp}^{u_{cs}} u_{cs}^{v_{cs}} = \alpha^{h(ID_{cs})} \bmod p$, CS accepts the signature (u_{cs}, v_{cs}), otherwise it rejects.

4.3. Pseudonym Authentication

①EV_i chooses an integer k at random from the range [1, p – 2] where $\gcd(k, p - 1) = 1$. To encrypt the message m, $d = T_{ev} \| m \| M_{32}$, calculate $w = \alpha^k \bmod p$,$z = d y_{cs}^{k} \bmod p$, where M_{32} is the last 32 bits of the key x_{ev}, T_{ev} is the timestamp. Then EV_i sends w and z to CS.

②CS use w, z decryption d, $d = w^{-x_{cs}} z \bmod p$, in order to obtain information $d = T_{ev} \| m \| M_{32}$. The CS verifies its identity by computing $m = C_1^{-x_{cs}} C_2 \bmod p$, where C_1 and C_2 are the values entered into the CS database upon registration. If the received m matches that of the SP data center, CS creates a timestamp T_{cs} and verifies that the timestamp T_{ev} is valid before accepting or rejecting the request. CS obtains m from d and examines if the received m is the same as that of the SP data center:

$$T_{cs} - T_{ev} < T \tag{3}$$

Where T is the delay time between CS and EV_i.

Then, CS picks an arbitrary number $k' \in [1, p - 2]$, figures out $r_{cs}' = \alpha^{k'} \bmod p$, and CS transmits r_{cs}' to EV_i.

③EV_i chooses two random numbers β_1, β_2, and uses these two parameters to blind r_{cs}' :

$$r_{cs} = d\alpha^{\beta_1} r_{cs}'^{\beta_2} \bmod p \tag{4}$$

$$a = r_{cs}\beta_2^{-1} \bmod (p - 1) \tag{5}$$

EV_i sends a to CS;

④$S_{cs}' = (k' + ax_{cs}) \bmod (p - 1)$ is computed by CS and sent to EV_i.

⑤EVi obtains $S_{cs} = (S_{cs}'\beta_2 + \beta_1) \bmod (p - 1)$ and checks whether $\alpha^{-S_{cs}} y_{cs}^{r_{cs}} r_{cs} = d \bmod p$ holds. If it is true, the final EV_i output digital signature (r_{cs}, S_{cs}).

4.4. Construct a Bloom filter

A Bloom filter is distributedly constructed using pseudonyms in this process. We designate a blockchain for the building of pseudonymous Bloom filters, and the CS public key r_{cs}' is accessible to all blockchain participants. So that pseudonyms belonging to an EV_i cannot be linked to each other. For each pseudonym and certificate, EVi creates a separate transaction. To sign the transaction data, a fresh set of transaction public and private keys (t_{pk}, t_{sk}) is created for every transaction (d, S_{cs}).

1) Users create their own private key and public key pair for transactions (t_{pk}, t_{sk}).

2) To transact with the content, the user chooses a pseudonym and signs the transaction using the private key tsk (d, S_{cs}).

3) The user broadcasts the transaction containing (d, S_{cs}) to miner, miner nodes are voted by each node, and all CS have their own identities $\{CS_1, CS_2, \ldots\ldots, CS_n\}$ participates in all node voting as the pre-selected node. A node with adequate hardware and robust communication capabilities is often chosen to cast a vote. All nodes continue to vote for the replacement primary node if the primary node fails.

4.5. Pseudonym Verification

The entity scans the final block of the blockchain for Bloom filters in order to quickly and effectively confirm the legitimacy of pseudonyms. The transaction containing the pseudonym must be located in order to complete the verification, and the certificate must be confirmed using the CS's public key.

①Initialization of Bloom filter: Bloom filter startup consists of originally setting each element in an array of q elements to zero. When we are provided a pseudonym ID, we use k hash functions to compute its k hash values as index values and set the mapping value to 1 at the appropriate point in the array. Hi (ID) mod q is used to calculate the array's index value. If many hash values point toward a single place, the value of the map won't rise.

②Bloom filter authentication: When a pseudonym's mapped value fulfills the requirement that each coefficient equate to hi (ID) mod q and the Bloom filter includes 0, the ID is invalid.

5. Conclusion

Before a person applies for a charging service, it is crucial to confirm their identity in order to safeguard their privacy from being violated. In this research, a blind signature-based distributed

privacy protection strategy for electric cars is proposed. This system provides a fully distributed environment to handle the problems of a central failure point and concentrated attack while guaranteeing user privacy. The suggested technique has less computational overhead while maintaining security, according to analysis.

References

[1] Liu Qilie, Chen Cheng. V2G anonymous Identity authentication scheme based on blockchain [J]. Computer Engineering, 2021, 47(11):7.

[2] Xia Z, Fang Z, Gu K, et al. Effective charging identity authentication scheme based on fog computing in V2G networks [J]. Journal of Information Security and Applications, 58, 2021, Article 102649.

[3] Chen J, Zhang Y, Su W. An anonymous authentication scheme for plug-in electric vehicles joining to charging/discharging station in vehicle-to-Grid (V2G) networks [J]. China Communications: English version, 2015(3): 11.

[4] Xu S, Chen X, He Y. EVchain: An Anonymous Blockchain-Based System for Charging–Connected Electric Vehicles [J]. Journal of Tsinghua University: Natural Science Edition (English Edition), 2021, 26(6):12.

[5] Wei K, Jian S, Pv C, et al. A practical group blind signature scheme for privacy protection in smart grid [J]. Journal of Parallel and Distributed Computing, 2020, 136:29-39.

[6] Weijian Zhang, Zhimin Guo, Nuannuan Li, Mingyan Li, Qing Fan, Min Luo, "A Blind Signature-Aided Privacy-Preserving Power Request Scheme for Smart Grid", Wireless Communications and Mobile Computing, vol. 2021, Article ID 9988170, 10 pages, 2021.

[7] Guan Z, Si G, Zhang X, et al. Privacy-preserving and Efficient Aggregation based on Blockchain for Power Grid Communications in Smart Communities [J]. IEEE Communications Magazine, 2018, 56(7):82-88.

[8] J. S. Shin, S. Lee, S. Choi, M. Jo and S. -H. Lee, "A New Distributed, Decentralized Privacy-Preserving ID Registration System," in IEEE Communications Magazine, vol. 59, no. 6, pp. 138-144, June 2021, doi: 10. 1109/ MCOM. 011.2000699.

[9] Popescu Constantin. A secure and efficient payment protocol based on ElGamal cryptographic algorithms [J]. Electronic Commerce Research, 2017.