SCIENCE CHINA
Information Sciences

• LETTER •

March 2023, Vol. 66 139102:1–139102:2
https://doi.org/10.1007/s11432-021-3378-3

DBKEM-AACS: a distributed key escrow model in blockchain with anonymous authentication and committee selection

Axin XIANG[1,2], Hongfeng GAO[1,3,4*], Youliang TIAN[1,2,4*] & Liang WAN[1,4,5*]

[1]*College of Computer Science and Technology, Guizhou University, Guiyang* 550025*, China;*
[2]*Institute of Cryptography and Data Security, Guizhou University, Guiyang* 550025*, China;*
[3]*Network and Information Management Center, Guizhou University, Guiyang* 550025*, China;*
[4]*State Key Laboratory of Public Big Data, Guizhou University, Guiyang* 550025*, China;*
[5]*Institute of Computer Software and Theory, Guizhou University, Guiyang* 550025*, China*

Received 18 August 2021/Revised 11 November 2021/Accepted 1 December 2021/Published online 14 November 2022

Citation Xiang A X, Gao H F, Tian Y L, et al. DBKEM-AACS: a distributed key escrow model in blockchain with anonymous authentication and committee selection. Sci China Inf Sci, 2023, 66(3): 139102, https://doi.org/10.1007/s11432-021-3378-3

Dear editor,

Key security is of great practical significance and demand to guarantee the security of digital assets in the blockchain system. At present, users prefer to escrow their assets on centralized institutions, but this phenomenon has gone against the unique characteristics of decentralization and anonymity in the blockchain. Among them, the suspense incidents of assets lock or lost are enough to prove that the security of escrowed keys on the exchanges is questionable.

Based on the above problem, many scholars put forward the idea of combining threshold cryptography with key escrow. Moreover, the dynamic committee proactive secret sharing scheme (CHURP) [1] holds the function of updating periodical shares. However, there is still a lack of research on committee selection, anonymous authentication, and instantiation. The main contributions are as follows: (1) We propose an appropriate committee selection algorithm. (2) We propose an anonymous authentication scheme suitable for the blockchain system. (3) We construct a distributed key escrow scheme that can achieve anonymous authentication and committee selection.

The proposed model. The DBKEM-AACS is shown in Figure 1. Its main idea is as follows. Firstly, escrow node (EN) embedded physically unclonable functions (PUF) achieves the selection of key escrow committees, such as the old committee (OC) and the new committee (NC). Secondly, EN realizes identity authentication between EN and smart contract (SC). Finally, EN generates, distributes, verifies, and updates the key sub-shares, and then recovers the key. Some preliminaries are provided in Appendix A. The pseudocodes for all contracts are provided in Appendix B.

Committee node selection algorithm. Now, we design an appropriate committee node selection algorithm to deal with the lack of a committee selection method in CHURP [1] by referring to [2]. Especially, the randomness and positivity of nodes are guaranteed by the activeness and the honest behavior of nodes is restricted by deposits and rewards.

(1) Defining activeness. (a) Initializing. If the blockchain node (BN_i) only participates in the transaction, the activeness calculation contract (ACC) computes an activeness $A_i(e) = \min(0.5 + tx_i/tx, 1)$. If ($\mathrm{BN}_i$) only participates in mining, ACC computes $A_i(e) = \min(0.5 + wk_i/wk, 1)$. If ($\mathrm{BN}_i$) do both, ACC computes $A_i(e) = \min(0.5 + wk_i/wk{+}tx_i/tx, 1)$. If BN_i do not do either, ACC computes $A_i(e) = 0.5$. (b) Updating activeness. If BN_i participates in the key escrow action during epoch e, then computes Eq. (1). If BN_i does not participate in the key escrow action during epoch e, then computes Eq. (2).

$$A_i(e{+}1){=}\begin{cases}\min\left(1, \left(1+\dfrac{1}{e}\right)A_i(e)\right), \\ \quad \text{if node submits correct key subshares;} \\ \max\left(0, \left(\dfrac{1}{1+e}\right)A_i(e)\right), \\ \quad \text{if node does not submit key subshares;} \\ 0, \text{ if node submits incorrect key subshares.}\end{cases} \tag{1}$$

$$A_i(e{+}1){=}\begin{cases}\min\left(1, A_i(e)+\left|\dfrac{wk_i}{wk}\right|\right), \\ \quad \text{if node participates in mining;} \\ \min\left(1, A_i(e)+\left|\dfrac{tx_i}{tx}\right|\right), \\ \quad \text{if node participates in transaction;} \\ \min\left(1, A_i(e)+\max\left(\left|\dfrac{wk_i}{wk}\right|, \left|\dfrac{tx_i}{tx}\right|\right)\right), \\ \quad \text{if node par ticipates in both mining} \\ \quad \text{and transaction;} \\ 0, \text{ if node submits incorrect key subshares.}\end{cases} \tag{2}$$

* Corresponding author (email: hfgao@gzu.edu.cn, youliangtian@163.com, wanliangtr@163.com)

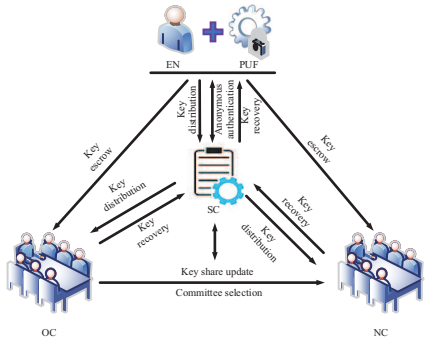

Figure 1 (Color online) The architecture of the DBKEM-AACS.

(2) Committee node selection contract (CNSC). (a) Initializ-ation. EN inputs the deposit d_{en}, committee size N, public key pk_{en} and reward r, and each candidate node BN_i inputs $A_i(e)$, pk_i and $d_{\text{BN}i}$ to CNSC. (b) Node sorting. After CNSC judges that its account balance is equal to $d_{\text{en}} + n \times d_{\text{BN}i}$, it ranks BN_i according to $A_i(e)$, and storing the sorted result in the pre-committee node directory (PCND). (c) Committee selection. CNSC selects N nodes from PCND and inputs them into the committee node directory (CND), and then outputs the remaining nodes in PCND to the eliminated node directory (END) and returns their deposits or rewards.

Anonymous authentication scheme based on PUF. The purpose is to solve the problem of the key ownership by the node in the blockchain system. The designed node authentication contract (NAC): (a) Setup. Making some parametric assumptions that there is a secure channel, a constant k_i, and a hash function H. (b) Login. EN_i calculates $d_i = H(N_i)\|\text{pk}_i\|(H(r_i) \oplus N_i)$ using PUF's return r_i, nonce N_i and H, and then inputs d_i to NAC. (c) Grant. EN_i calculates $\text{GM}_i = d_i\|r_i \oplus m_i\|H(r_i \oplus m_i)\|\text{“Grant”}\|k_i$ after XORing a content m_i with r_i, and then inputs GM_i to NAC. (d) Authentication. After EN_i enters $H(r_i)$ into NAC, it verifies $k_i' \leqslant k$ and N_i in d_i by $H(r_i)$, searches and sends GM_i to EN_i, and then calculates $k_i' = k_i' + 1$ and public on the blockchain. Finally, EN_i verifies the correctness of $r_i \oplus m_i$ using H, and then obtains m_i by XORing $r_i \oplus m_i$ with r_i. (e) Revocation. if $k_i' > k_i$ is true, NAC publishes the message $d_i\|r_i \oplus m_i\|\text{“Revoke”}\|k_i'$ on blockchain to revoke EN_i's access permission to $r_i \oplus m_i$.

Distributed key escrow scheme. Now, we instantiate the theoretical idea of the "dimension-switching" [1] by specifying and disclosing one type of $\langle t, 2t\rangle$-order binary polynomial: $B(x, y) = s + a_1x + a_2x^2 + \cdots + a_tx^t + b_1y + b_2y^2 + \cdots + b_{2t}y^{2t} + cxy$. The execution items of the proposed key escrow contract (KEC) are as follows: (1) Initialization. calling CNSC to select OC: $C_i^{(e)} = \{P_1^e, P_2^e, \ldots, P_n^e\}$ and NC: $C_i^{(e+1)} = \{P_1^{e+1}, P_2^{e+1}, \ldots, P_n^{e+1}\}$, calling NAC to complete the anonymous authentication of EN_i to the key sub-share $B_i(h, y)$. (a) $B_i(h, y)$ generation. EN_i secretly constructs $B_i(x, y) = \text{sk}_i + a_1^i x + a_2^i x^2 + \cdots + a_t^i x^t + b_1^i y + b_2^i y^2 + \cdots + b_{2t}^i y^{2t} + c^i xy$. EN_i calculates $B_i(h, y)$ for each node P_h^e in $C_i^{(e)}$ and $H(\text{sk}_i)$, where $h = 1, 2, \ldots, n$. (b) $B_i(h, y)$ distribution. after EN_i securely inputs all $B_i(h, y)$, $H(\text{sk}_i)$ and pk_i into KEC, it calculates the secure sub-secret share $c_i^h = B_i(h, y) \oplus r_h$ using P_h^e's r_h, $s_h = c_i^h\|H(c_i^h)\|\text{pk}_h\|\text{pk}_i$ and publish s_h on the blockchain together with $H(\text{sk}_i)$. (c) $B_i(h, y)$ verification. P_h^e retrieves s_h from the blockchain and verifies the correctness of s_h by using H, c_i^h, and $H(c_i^h)$, and then obtains $B_i(h, y)$ by calculating $r_h \oplus c_i^h$. (d) $B_i(h, y)$ update (some details are provided in Appendix C). Firstly, KEC, P_h^e, each node in $U' = \{C_i^{(e+1)}\}_{j\in\{2t+1\}}$ together construct $B_i(x, j)$ to achieve share reduction [1] based on Eq. (3) and $t+1$ $B_i(h_k, j)$, where a_i, $c(h_k)j$, $f(j)$ are all unknown. Secondly, U_j' calculates $B_i'(x, j) = B_i(x, j) + Q_i(x, j)$ to achieve share proactivization [1] based on a 0-shared polynomial $Q_i(x, j)$.

$$\begin{cases} a_0 + a_1h_1 + \cdots + a_t(h_1)^t + f(j) + c(h_1)j = B_i(h_1, j), \\ a_0 + a_1h_2 + \cdots + a_t(h_2)^t + f(j) + c(h_2)j = B_i(h_2, j), \\ \qquad\qquad \vdots \\ a_0 + a_1h_{t+1} + \cdots + a_t(h_{t+1})^t + f(j) + c(h_{t+1})j \\ \quad = B_i(h_{t+1}, j). \end{cases} \tag{3}$$

Finally, P_h^{e+1} in $C_i^{(e+1)}$ constructs $B_i'(h, y)$ to achieve full-share distribution [1] based on Eq. (4) and $2t + 1$ $B_i'(h, j_k)$, where e_i, $c(h)j_k$, $g(h)$ are all unknown.

$$\begin{cases} e_0 + g(h) + e_1j_1 + \cdots + e_{2t}(j_1)^{2t} + c(h)j_1 = B_i'(h, j_1), \\ e_0 + g(h) + e_1j_2 + \cdots + e_{2t}(j_2)^{2t} + c(h)j_2 = B_i'(h, j_2), \\ \qquad\qquad \vdots \\ e_0 + g(h) + e_1j_1 + \cdots + e_{2t}(j_{2t+1})^{2t} + c(h)j_{2t+1} \\ \quad = B_i'(h, j_{2t+1}). \end{cases} \tag{4}$$

(e) Key recovery. EN_i constructs $B_i'(x, y)$ based on Eq. (5) and $t + 1$ P_h^{e+1}'s $B_i'(h_k, y)$, where sk_i, m_v, $c(h_k)$, $f(y)$ are all unknown but each $f(y) + c(h_k)y$ is known, and $t + 1$ is the threshold value.

$$\begin{cases} \text{sk}_i + m_1h_1 + \cdots + m_t(h_1)^t + f(y) + c(h_1)y = B_i'(h_1, y), \\ \text{sk}_i + m_1h_2 + \cdots + m_t(h_2)^t + f(y) + c(h_2)y = B_i'(h_2, y), \\ \qquad\qquad \vdots \\ \text{sk}_i + m_1h_{t+1} + \cdots + m_t(h_{t+1})^t + f(y) + c(h_{t+1})y \\ \quad = B_i'(h_{t+1}, y), \end{cases} \tag{5}$$

and then EN_i recovers $\text{sk}_i = B_i'(0, 0) = B_i(0, 0)$.

Security analysis and performance analysis are provided in Appendixes D and E, respectively.

Acknowledgements This work was supported by National Natural Science Foundation of China (Grant Nos. 61662009, 61772008), Science and Technology Major Support Program of Guizhou Province (Grant No. 20183001), Key Program of National Natural Science Union Foundation of China (Grant No. U1836205), Science and Technology Program of Guizhou Province (Grant No. [2019]1098), Project of High-level Innovative Talents of Guizhou Province (Grant No. [2020]6008), and Science and Technology Program of Guiyang (Grant No. [2021]1-5).

Supporting information Appendixes A–E. The supporting information is available online at info.scichina.com and link.springer.com. The supporting materials are published as submitted, without typesetting or editing. The responsibility for scientific accuracy and content remains entirely with the authors.

References

1 Maram S K D, Zhang F, Wang L, et al. CHURP: dynamic-committee proactive secret sharing. In: Proceedings of ACM SIGSAC Conference on Computer and Communications Security, London, 2010. 2369–2386
2 Lei K, Zhang Q C, Xu L M, et al. Reputation-based Byzantine fault-tolerance for consortium blockchain. In: Proceedings of the 24th IEEE International Conference on Parallel and Distributed Systems, Singapore, 2018. 604–611