SCIENCE CHINA
Information Sciences

• LETTER •

April 2018, Vol. 61 049102:1–049102:3
https://doi.org/10.1007/s11432-017-9245-1

DESC: enabling secure data exchange based on smart contracts

Jiao LIANG[1], Weili HAN[1*], Zeqing GUO[1], Yaoliang CHEN[1], Chang CAO[1], Xiaoyang Sean WANG[1] & Fenghua LI[2]

[1]*Software School, Fudan University, Shanghai* 201203*, China;*
[2]*Institute of Information Engineering, Chinese Academy of Sciences, Beijing* 100093*, China*

Received 6 August 2017/Accepted 20 September 2017/Published online 8 February 2018

Citation Liang J, Han W L, Guo Z Q, et al. DESC: enabling secure data exchange based on smart contracts. Sci China Inf Sci, 2018, 61(4): 049102, https://doi.org/10.1007/s11432-017-9245-1

Dear editor,

As data are one of the most important factors of today's intelligent systems but are relatively scarce, people try exchanging data with other organizations or public marketplaces. However, data exchange is now lack of a secure execution mechanism to automatically and fairly protect the rights which are seriously concerned by data owners. For instance, data owners cannot limit the identity of a data transferee who is banned to take the data by any way. Moreover, data transferees are also afraid that their applications for exchange may be unfairly treated. For instance, data owners could dynamically change data price when multiple transferees submit their applications at the same time. A data marketplace can take over the role of a trusted third party during data exchange, but single point failure is always a serious problem in a large-scale distributed system. Furthermore, these marketplaces usually use contracts based on natural languages which may be maliciously misinterpreted and cannot be automatically executed.

Numerous researches have been presented to provide digital rights management (DRM) [1]. For instance, Zhu et al. [2] presented a privacy-preserving video subscription scheme with the limitation of expire date. However, rights of data owners are more complex than rights of digital content owners. Decentralized and trustworthy enforcement for access control policies is a long-term issue. Han et al. [3] proposed an optimized mechanism which is a trusted and decentralized access control framework for the client/server architecture. These studies still need third party authorities which are used to execute or supervise the rights control, leading to a few trust and security issues. Existing studies [4] on attribute-based access control have been applied to many fields, such as e-commerce and Internet of Things. This article introduces attribute-based access control and proposes a secure framework for data exchange based on smart contracts (DESC) to enable automatic and fair access control. Featuring with decentralized, fair, transparent, immutable and traceable advantages, blockchain 2.0 [5] provides a generalized framework for implementing decentralized compute resources named smart contracts which define rights and policies in mathematical and programming forms. Once the predefined smart contract is triggered by a transaction, it can automatically execute the specific contractual clauses.

To find out which data rights are seriously concerned by data owners, we investigate the DRM mechanisms and several data exchange platforms: GBDEX[1)], national engineering lab for big data distribution and exchange technologies[2)]. Then,

* Corresponding author (email: wlhan@fudan.edu.cn)

1) http://www.gbdex.com/website/.
2) http://www.chinadaily.com.cn/china/2017-03/11/content_28519995.htm.

we articulate each essential data right which may be maliciously used or misused as follows.

(1) Blacklist right. A data owner has the right to limit the identities of transferees who are denied to access the data by any way.

(2) Whitelist right. A data owner could specify some transferees who are allowed to access the data in any case.

(3) Price right. A data owner has the right to set the price of his or her data by specifying a price range for the data.

(4) Exchange time right. A data owner has the right to limit the period for data exchange by setting a valid time period.

(5) Sample rate right. A data owner could decide the amount of data that are used to exchange by sampling from the raw data. That is, the owner could set a valid sample rate interval for data.

(6) Update frequency right. A data owner could specify the valid update frequency by setting an update frequency range for the data.

(7) Liquidated damages right. A data owner could set a liquidated damages range to reassure the validity of an exchange application. When a transferee applies for exchange, he or she should pay for the liquidated damages. If the damages are out of the range or the application has been denied by other policies, the transferee will receive a full refund and is denied to access the data. Otherwise, the liquidated damages may be kept in the account of smart contract until the owner or a data marketplace dominate it after exchange time which is stated by the transferee. That is, the owner may charge the liquidated damages if the transferee violates exchange attributes, e.g., the transferee who has gained a license does not exchange data at the exchange time. Otherwise, the damages will be returned to the transferee after successful exchange.

We conclude the following three assumptions for DESC framework.

(1) Each legitimate data exchange application must be enforced through DESC framework.

(2) Each participant of DESC must pass an identity authentication which is held by a trusted authority, e.g., data marketplace.

(3) Data can only be exchanged by the unique data owner who is responsible to store the data, while transferees who have acquired the data via DESC are merely data possessors.

DESC framework. The workflow of DESC framework is shown in Figure 1, where the three nodes represent one data owner and two transferees, and the smart contract denotes the key part of the proposed DESC framework. The main contents of the DESC framework are as follows.

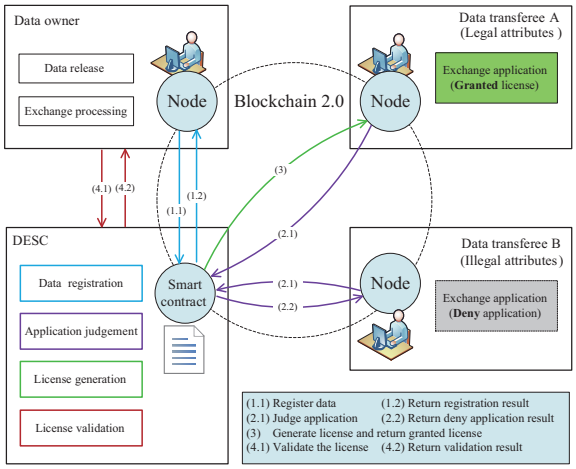

Figure 1 (Color online) DESC framework workflow.

(1) Data registration. The data owner creates a set of data rights policies, and submits data attributes to DESC for data registration, then DESC registers the data and returns the registration result. The data attributes consist of the unique data identifier, the unique identity of data owner, data description and a set of policy attributes which is created by the owner to protect his or her data rights. Each element in policy attributes represents a constraint condition, e.g., price constraint denotes a price range for data.

(2) Application judgement. To apply for data exchange, two data transferees (transferees A and B) submit their exchange attributes: transferee's identity, data identifier, the price that the transferee will pay for the data, the time when the data will be exchanged. The components of exchange attributes are corresponding to exchange policies. Then, DESC judges whether the transferee is allowed to exchange data. Figure 1 shows that the transferee A is legal to access the data, while transferee B is not. Finally, DESC returns deny application to transferee B, and the application result for transferee A moves to the next step of license generation.

(3) License generation. When transferee A is legal to access the data, DESC generates a license, and returns the granted license to the transferee A. The license can be performed as a hash value by leveraging a cryptographic hash function. The components of a license include data owner's identity, data identifier, exchange attributes created by transferee A and unique identifier (e.g., blockchain address) of the smart contract.

(4) License validation. When a user takes a license to exchange data, the data owner could submit the use's license and identity to DESC to validate the license, DESC then returns validation result by tracing blockchain transaction records.

Access control enforcement. Algorithm 1 shows the pseudocode of policy enforcement algorithm. If a transferee is allowed to access the data, DESC will grant a license via genLicense() method. Otherwise, DESC will return deny application with a full refund of liquidated damages. The policy enforcement of DESC performs policy combination, where the policy enforcement sequence is: blacklist policy → whitelist policy → price policy → exchange time policy → sample rate policy → update frequency policy → liquidated damages policy. If a transferee's identity is in blacklist, then the transferee is denied to access the data without the executions of other policies. Otherwise, the transferee is granted access by the blacklist policy and move to the execution of whitelist policy. If the identity is in whitelist, then the transferee will be granted to access the data without executions of the next five policies. If not, the rest five policies will perform deny override algorithm which will return deny application if at least one of them is denied access.

Algorithm 1 Policy enforcement algorithm

Input: ea: exchange attributes; pa: policy attributes.
Output: result: application result for data exchange.

```
for identity ∈ pa.blacklist do
    if ea.identity = identity then
        result ← "deny application";
        Return result;        ▶ deny application
    end if
end for
for identity ∈ pa.whitelist do
    if ea.identity = identity then
        result ← genLicense();
        Return result;        ▶ grant license
    end if
end for
if ea.price ∈ pa.priceRange and
      ea.time ∈ pa.exchangeTimeRange and
      ea.sample ∈ pa.sampleRateRange and
      ea.update ∈ pa.updateFrequencyRange and
      ea.damages ∈ pa.liquidatedDamagesRange then
    result ← genLicense();
    Return result;          ▶ grant license
end if
result ← "deny application";
Return result.              ▶ deny application
```

Evaluation. We implemented two prototypes and deployed them on Private Ethereum [6] and Hyperledger Fabric [7], respectively. Because the automation and fairness of DESC could be guaranteed by blockchain 2.0 technology, we evaluated time performance of exchange application, and the results are 2.48–5.60 ms on Private Ethereum and 0.85–4.15 ms on Fabric. Specifically, the time consumption mainly depends on the processes of license generation as well as identity comparison in blacklist policy and whitelist policy.

Discussion. Due to the inherent publicity of blockchain, policies can be inevitably seen by public. The proposed DESC framework has not considered policy confidentiality yet, but it can be partly mitigated by using existing cryptography technologies, such as salted hash [8] and order preserving encryption (OPE) [9]. Moreover, license forgery could be mitigated by our DESC framework because license validation could verify whether the user is legitimate to access the data by tracing immutable blockchain transaction records.

Conclusion. To the best of our knowledge, this is the first work to leverage smart contracts on blockchain 2.0 to automatically and fairly control the access in the domain of secure data exchange. We articulated seven data rights and implemented our proposed DESC framework by expressing data rights policies in smart contracts. After conducting experiments on Private Ethereum and Hyperledger Fabric, we concluded that the proposed DESC framework is promising to automatically and fairly protect data owners' rights. Moreover, the experimental results indicate that access control based on smart contracts has a broad prospect.

Acknowledgements This work was supported by National Natural Science Foundation of China (NSFC) (Grant No. 61572136), and Shanghai Innovation Action Project (Grant No. 16DZ1100200).

References

1 Ma Z F. Digital rights management: model, technology and application. China Commun, 2017, 14: 156–167
2 Zhu L H, Chen M X, Zhang Z J, et al. A privacy-preserving video subscription scheme with the limitation of expire date. Sci China Inf Sci, 2017, 60: 098101
3 Han W L, Xu M, Zhao W D, et al. A trusted decentralized access control framework for the client/server architecture. J Netw Comput Appl, 2010, 33: 76–83
4 Hu V C, Kuhn D R, Ferraiolo D F. Attribute-based access control. Computer, 2015, 48: 85–88
5 Kosba A, Miller A, Shi E, et al. Hawk: the blockchain model of cryptography and privacy-preserving smart contracts. In: Proceedings of IEEE Symposium on Security and Privacy (SP), San Jose, 2016. 839–858
6 Buterin V. Ethereum: a next-generation smart contract and decentralized application platform. 2017. https://github.com/ethereum/wiki/wiki/White-Paper
7 Androulaki E, Cachin C, Christidis K, et al. Hyperledger fabric proposals: next consensus architecture proposal. 2017. http://github.com/hyperledger/fabric/blob/master/proposals/r1/Next-Consensus-Architecture-Proposal.md
8 Kent A D, Liebrock L M. Secure communication via shared knowledge and a salted hash in ad-hoc environments. In: Proceedings of Computer Software and Applications Conference Workshops, Munich, 2011. 122–127
9 Boldyreva A, Chenette N, O'Neill A. Order-preserving encryption revisited: improved security analysis and alternative solutions. In: Proceedings of the 31st Annual Conference on Advances in Cryptology, Santa Barbara, 2011. 578–595