Chaincode-based Access Control System for Multi-Administrative Domains

Sherzodbek Abdurahmanov, Beomseok Kim, Ki-Hyung Kim
Ajou Univ.

Abstract

Access Control Services are recently have become increasingly popular as independent service providers according to the Software as a Service (SaaS) model. The more popularity they gain the more inclined they become for threats and attacks. On the other hand, the blockhchain technology affected a great wonder with its security techniques. Being motivated of these technologies, we moved our new approach idea forward to combine and leverage the advantage techniques of both technologies. In our research, we conducted a new approach for carrying out the privacy protection in Attribute-based Access Control Services. So as to perform the proposed idea we use a permissioned blockchain platform namely, Hyperledger Fabric since it supports both the private and public data collections. In a proposed idea the attributes and policies are stored using ledger's both public and private data storages. Meantime, almost all components of Access Control system are written as chaincodes (smart contracts) and are used to add and delete users, to store attributes and policies and to make decisions either giving or denying the request.

Ⅰ. Introduction

Our strategy approach is based on the regulations of ABAC. An ABAC protocol consisting of a bunch of rules implemented using a proper combination of algorithms (logical operations such as IF, AND, OR, and others) and must therefore be followed in order to provide the necessary entry. We use XACML for policy writing because it is a very concise language that allows complicated ABAC policies to be written and reusable for further implementations [1].

1.1 XACML standard

The XACML stands for "eXtensible Access Control Markup Language" and is basically an ABAC framework where attributes relating to a user or action or feature are added to decide whether a particular user should have special access to a certain resource [2]. To be more familiar with XACML, the roles of the components are briefly described:

- Policy Enforcement Point (PEP) is the element combined with the protected property, and is capable of intercepting request for entry, initiating the decision process, and executing the relevant outcome by enabling or refusing access execution.
- Policy Administration Point (PAP) is the element that controls the policy of access authorization.
- Attribute Managers (AMs) are the elements that handle the subject and object attributes along with environment attributes, enabling their values to be retrieved and modified.
- Policy Information Points (PIPs) is for addition, different Attribute Managers control the set of characteristics needed for the policy assessment.
- Policy Decision Point (PDP) is the validation mechanism that retrieves a policy written by resource owner, an entry request, and the entity of attributes as data, tests the proposal and sends back the decision (allowed or denied).

1.2 Related works

As much as it is known, many proposals have been submitted for blockchain-related access control schemes. The authors in [3] collected all blockchain based Access Control research where the most closely related research to ours is [4], which introduces the Ethereum blockchain attribute-based access control (ABAC) policies. This poses three main differences in our work: (i) it requires every participant of system to have GAS wallet in order to make transactions, (ii) it allows access control privileges to be shared between users, and (iii) it considers to use a public blockchain, unlike our private transaction approach.

Ⅱ. Chaincode-based Access Control System

The main idea of our solution is to create a Hyperledger Fabric-based access control mechanism for companies to share resources with each other, where we use Hyperledger Fabric to produce an authorization request, store and retrieve access control rules, attributes and complete the access decision process. In contrast of other proposed systems, in our architecture we propose a new approach where we set a Channel composed of Peers (PIPs in our system), which are utilized to store the attributes and policies on ledgers as a chaincode while the Policy Decision Point (PDP) is also installed to peers (PIPs) and in case of necessity the endorsement policy for the channel will be overridden with additional rules in agreement of the channel members which are companies, sharing their resources with each other, in our use case.

2.1 Policy creation

The policy development is the first phase of the life cycle of the system. First, the resource provider creates an XACML code that uses the WSO2

Identity Server tool to establish the resource's access rights and it offers a simpler way to describe the rules and construct a framework for XACML 3.0 standard.

Afterwards, those policies are stored onto the ledger and this is accomplished by deploying a chaincode to the channel. Recently, three programming languages can be used in Fabric chaincode including Golang, Node.js and Java. To deploy chaincode, a resource owner must install the chaincode onto target peer (PIP) and then invoke an Ordering service (PAP) to instantiate the chaincode onto the channel. Accordingly, the PAP is invoked with the task of performing a commit of the chaincode to the channel.

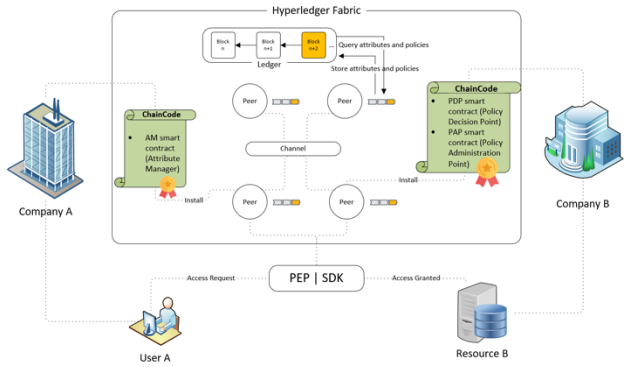

Figure 2. Proposed architecture

2.2 Storing attributes

After installing the chaincodes to the peers, now, it is time to store attributes and policies onto the ledgers invoking the chaincodes. The chaincode is written same as outlined in the previous chapter, either by the authorization service or by the trusted administrative domains. It is possible because in Hyperledger Fabric, writing more than one chaincode on the same channel or in Peer nodes is allowed and the chaincodes written on the same channel or on the member Peers of that channel enable to read and write operations to each other, whereas the chaincode of a different channel is only allowed to read the chaincode of another channel and the channel member, Peer node.

2.3 Access request time

Once the policies have been created and stored with the attributes onto the ledgers of Peers, the Access Control service based on Hyperledger Fabric begins to wait for requests for entry. The Policy Enforcement Point (PEP) is inserted in the code component that implements the mechanism to access the secured property so that each new access request is intercepted. As an application, this element is responsible for bridging the request of the user with the Hyperledger Fabric. Therefore, before disseminating the specification to the peers, the PEP draws up a transaction proposal using a licensed SDK (developed in Java) that uses one of the available APIs to produce a transaction proposal. The idea is a specification for a PDP chaincode feature to be invoked with certain input parameters such as user attributes in order to gain access. The SDK acts as a shim for the correctly constructed assembly of the contract application. The PDP chaincode is then invoked to get the policies committed earlier and the stored attributes to generate transaction outcomes including an answer tag (approve/deny). The response value for the request is then passed back to the PEP giving to the user either permission or denial accordingly.

III. Conclusion

We have presented and implemented, in this research, a new architecture for multi-parties to support an Access Control Service based on blockchain smart contract. In contrast of other existed architectures, our design represents a new approach leveraging the advantage techniques of Hyperledger Fabric, to protect data privacy throughout the public network. Additionally, in our proposed schema the members of the network cannot see the Statuses of other members whether they were granted or denied having access to the server. Moreover, our proposed access control service is capable to reduce the possibility of giving access to a wrong user, since Attribute-based Access Control is a strong tool to recognize the right user. The permissioned blockchain deals out the network policies and aims to address the limitations of traditional approaches while preserving each organization's independence.

ACKNOWLEDGMENT

본 연구는 과학기술정보통신부 및 정보통신기획평가원의 대학 ICT 연구센터지원사업의 연구결과로 수행되었음" (IITP-2020-2018-0-01396)

이 연구는 2020 년 정부(과학기술정통부)의 재원으로 정보통신기술진흥센터의 지원과 2020 년 정부(산업통상자원부)의 재원으로 한국산업기술진흥원의 지원을 받아 수행된 연구임 (P0008703 2020 년 4 차 산업혁명 기반 산업기술보호 R&D 전문인력 양성)

Reference

[1] R. Guo, H. Shi, Q. Zhao, D. Zheng, "Secure Attribute-Based Signature Scheme with Multiple Authorities for Blockchain in Electronic Health Records Systems", IEEE Access, vol. 6, pp. 11676-11686, 2018.

[2] OASIS: eXtensible Access Control Markup Language (XACML) version 3.0 (January 2013).

[3] Sara R., Ralph D. Blockchain based access control systems: State of the art and challenges. Web Intelligence '19, October 14- 17, 2019, Thessaloniki, Greece.

[4] Damiano M., Paolo M., Laura R.: Blockchain Based Access Control Services. IEEE Confs on Internet of Things, Green Computing and Communications, Cyber, Physical and Social Computing, Smart Data and Blockchain, 2018.