A TOTP−Based Two Factor Authentication Scheme for Hyperledger Fabric Blockchain

Woo-Suk Park, Dong-Yeop Hwang, Ki-Hyung Kim
Dept. of Knowledge Information Engineering, Computer Engineering, Cyber Security
Ajou University,
Suwon, Republic of Korea.
Pwsuk9004@ajou.ac.kr, bc8c@naver.com, kkim86@ajou.ac.kr

Abstract— In this paper, we propose a new authentication method to prevent authentication vulnerability of Claim Token method of Membership Service provide in Private BlockChain. We chose Hyperledger Fabric v1.0 using JWT authentication method of membership service. TOTP, which generate OTP tokens and user authentication codes that generate additional time-based password on existing authentication servers, has been applied to enforce security and two-factor authentication method to provide more secure services.

Keywords—Hyperledger Fabric; TOTP; JWT; Two-Factor Authentication; Membership service

I. Introduction

Recently, the blockchain is rapidly emerging as a distributed database technology. The blockchain called a distributed ledger is a technique to prevent hacking that can occur when trading with virtual currency [1]. Since the bit coin developed by Satoshi Nakamoto has received attention, much research is underway on how to incorporate it into various fields including the financial sector. A lot of research is underway with Hyperledger, which supports Membership service only for authorized users as a private block chain.

The blockchain can be broadly divided into public blockchains and private blockchains [2]. Bitcoin [3] and Etherium [4] are typical public block chains. A lot of research is underway with Hyperledger, which supports Membership service only for authorized users as a private block chain [5].

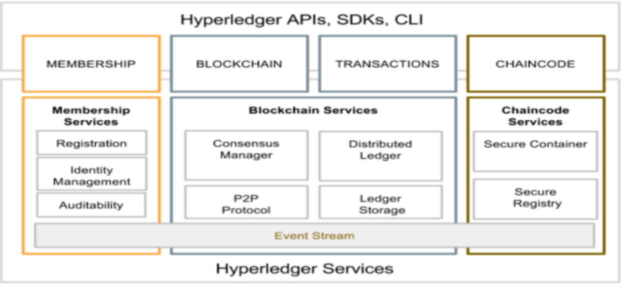


Fig. 1. Hyperledger Services

Hyperledger Fabric, which is a private block chain, has a structure that only authorized users can participate by membership service as shown in the figure 1 [6]. The membership service is a process of authenticating a member by performing registration, issuance, and verification of the access token between a user (client), a certification authority (CA), and a endorser (peer). As a membership service, the hyperledger uses the JWT (Json Web Token) authentication method based on the claim token as shown in the figure 2 [7].

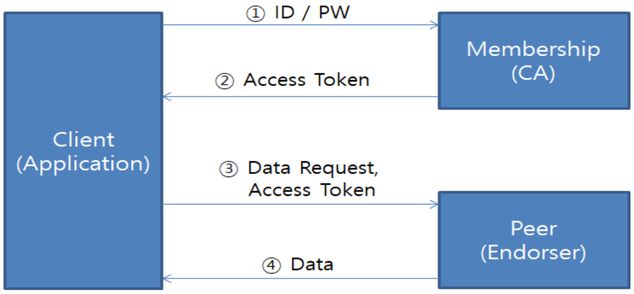


Fig. 2. Hyperledger Fabric v1.0 Authentication Process

However, the BASE64 is only used as the encoding method and data in claim tokens are not encrypted. Thus, malicious users could eavesdrop on access tokens of genuine users and could find out sensitive information [8][9]. To address this vulnerability, a strict method of registration, issuance, and verification of the access token is needed.

In this paper, we propose a two factor authentication scheme for hyperledger fabric blockchain using TOTP (time-based one-time password) as a authentication method that improved the JWT authentication method [10]. The TOTP generates a password using the current time information and the secret key shared by a TOTP server and a user. The TOTP-based two factor authentication scheme can guarantee a high level of authentication because it does not send the access token directly to the user.

The rest of this paper is organized as follows. We proposed a new method for a two factor authentication scheme for

978-1-5386-4646-5/18/$31.00 ©2018 IEEE

hyperledger fabric blockchain using TOTP in Section 2. We analyze the security of the proposed architecture in Section 3. Finally, we conclude this paper with future research direction in Section 4.

II. Proposal

In this chapter, we propose a secure authentication method to address the vulnerability of the JWT token authentication method used in Hyperledger Fabric.

A. Two Factor Authentication Scheme

Figure 3 shows the architecture of the TOTPS-based two factor authentication system based on the current authentication structure.

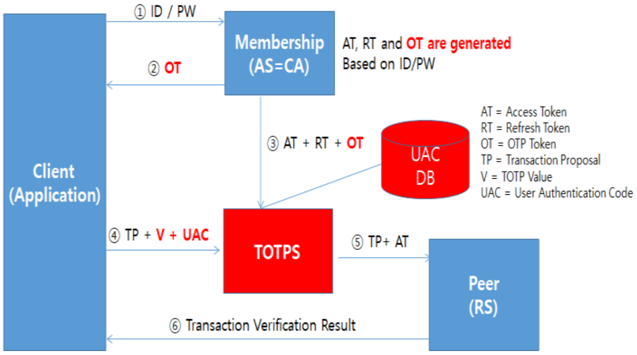


Fig. 3. The Proposed Two-Factor Authentication

When a client successfully logs in by entering a user ID and password, the member server generates AT and OT, sends OT to the client, and sends AT and OT to TOTPS, depending on the user's authority. The client calculates the TOTP password, V, by the OT received from the membership server and the current time, and transmits the transaction proposal to the TOTPS along with the TOTPS password and the user's personal authentication code (UAC) information issued by the TOTPS. TOTPS calculates the TOTP password V 'using the same OT from the membership server and compares it with the V value. If the V and V ' received from the client are the same, the user is first authenticated using the user authentication code. If authentication is successful, the client sends the proposed transaction and the AT it received from the member server to the peer. The peer performs the secondary authentication on the token issued by the membership server through the signature information of the endorser AT, checks the authority information defined in the AT, and transmits the transaction verification result to provide the security service to the client.

B. OTP Token Generation Structure

When the client logs in with the registered user ID and password, the membership server issues the access token and the OTP token through the authority set by the user when registering the member. The generated JWT token is composed of three parts: Header, Payload, and Signature. The Access Token issued is AT and the OTP Token is OT [6].

$$\text{AT} = \text{HMAC(Header, Payload, secret)} \tag{1}$$

$$\text{Payload} = [\text{ID, CA, T, …, 0}] \tag{2}$$

$$\text{OT} = \text{HMAC(Header, Payload, secret)} \tag{3}$$

$$\text{Payload} = [\text{ID, CA, T, …, 1}] \tag{4}$$

The header contains the hash algorithm and JWT token type information. The payload includes an identity (user ID) and a user certification authority (CA) exp (token expiration time). The last 0/1 is the flag information indicating the access token and the OTP token. The two tokens are created with the same payload, except for flag information, and the OTP token is available in a way that can be easily calculated from the access token information provided by the server.

C. User Authentication Code and TOTP Value Generation

According to RFC 4226, there are three steps to calculate the HOTP value [5]. First, Generate an HMAC-SHA-1 value Let HS = HMAC-SHA-1(K,C) HS is a 20-byte string Second, Generate a 4-byte string DT, defined below, returns a 31-bit string(Dynamic Truncation) Third, Compute an HOTP value Let Snum = StToNum(Sbits) Convert S to a number in 0...2^{31}-1 Return D = Snum mod 10^Digit. In the case of TOTP the incrementing counter is replaced by number of time-steps, i.e. (Unix Time-stamp-T) / X. Here T is the initial counter and X is the time-step [11].

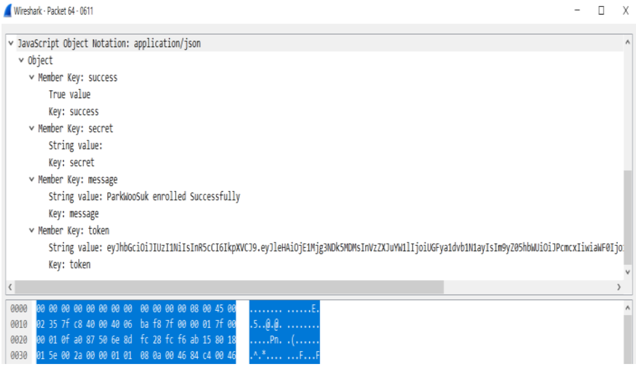
To generate an OTP password, the OTP password is generated from the same secret information that is shared by the OTP issuing server and individual users. Upon enrollment, TOTPS generates unique user authentication code information for the user. The user authentication code generated by TOTPS is used as information for identifying the user, and the user authentication code information is stored in the DB of the server. The secret to generate the password uses the OTP token issued by the member server.

A client that receives an OTP token on a member server uses the OTP token and the current time value to calculate the password V. The client sends the transaction proposal, the generated password V, and the user's personal authentication code to TOTPS.

TOTPS uses OTP Token issued by the Membership Server and the current time to generate V' and compares it with V sent from the client. If V and V' are the same, the user authentication code information is compared with the user of the access token and authenticated. Because the access token is not sent to the client, an attacker can not eavesdrop on the access token, and even if the OTP token is intercepted, the user's authentication code information can not be hacked.

III. Security Analysis

In this paper, we designed a two-factor authentication system to supplement important information disclosure and header defects of JWT token authentication method for membership service of existing Hyperledger fabric block chain network. Figure 4 shows the contents of the packet transmitted from the client and the authentication server as captured by the existing block-chain network.



```
∨ JavaScript Object Notation: application/json
  ∨ Object
    ∨ Member Key: success
        True value
        Key: success
    ∨ Member Key: secret
        String value:
        Key: secret
    ∨ Member Key: message
        String value: ParkWooSuk enrolled Successfully
        Key: message
    ∨ Member Key: token
        String value: eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJleHAiOjE1Mjg3NDk5MDMsInVzZXJuYW1lIjoiUGFya1dvb1N1ayIsIm9yZ05hbWUiOiJPcmcxIiwiaWF0Ijo
        Key: token
```

```
0000  00 00 00 00 00 00 00 00  00 00 00 00 08 00 45 00   ........ ......E.
0010  02 35 7f c8 40 00 40 06  ba f8 7f 00 00 01 7f 00   .5..@.@. ........
0020  00 01 0f a0 87 50 6e 8d  fc 28 fc f6 ab 15 80 18   .....Pn. .(......
0030  01 5e 00 2a 00 00 01 01  08 0a 00 46 84 c4 00 46   .^.*.... ...F...F
```

Fig. 4. Packet Information of Client and Membership Server

In an existing membership service, an attacker could exploit the packet information in Figure 4 to disguise itself as a user or tamper with the JWT token itself. However, the two-factor authentication scheme does not send the access token to the client, so the access token can not be eavesdropped. Even if the OTP token is eavesdropped, the access token can not be calculated because it does not know the secret information of the server. In addition, when a member is registered, the attacker can not obtain the user authentication code information generated by the TOTP server, so the TOTP server authentication fails and the unauthorized user can safely manage the data.

IV. Conclusion

The proposed architecture can not eavesdrop an access token because it does not send an authentication token from the authentication server to the client. In addition, since the user authentication code information can not be obtained, there is an advantage that the safety of the user can be assured even if the TOTP is calculated. However, from the performance point of view, there is a drawback that authentication execution time increases and transaction speed decreases. In future research, it is necessary to study how to solve the problem of TPS performance degraded by the authentication method through the members' consensus algorithm.

Acknowledgment

This research was supported by the MSIP(Ministry of Science, ICT & Future Planning), Korea, under the SW master's course of a hiring contract program (H0116-16-1006) supervised by the IITP(Institute for information & communications Technology Promotion) and This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (no. NRF-2018R1D1A1B07048697).

References

[1] Andreas M, Mastering Bitcoin: Unlocking Digital Cryptocurrencies, pp.49-68, O'REILLY, 2015

[2] Zibin Zheng, Shaoan Xie, Hongning Dai, Xiangping Chen, Huaimin Wang, "An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends", IEEE 6th international Congress on Big Data, 2017

[3] Satoshi Nakamoto, Bitcoin: A Peer-to-Peer Electronic Cash System, 2008.

[4] The Cointelegraph. A Brief History of Ethereum From Vitalik Buterin's Idea to Release, 2015

[5] E Androulaki, A Barger, V Bortnikov, "Hyperledger Fabric: A Distributed Operating System for Permissioned Blockchains", EuroSys'18, April 23-26, Porto, Portugal

[6] Sousa J, Bessani A, Vukolic M, "A byzantine fault-tolerant ordering service for the hyperledger fabric blockchain platform" arXiv:1709.06921, 2017

[7] Jones Michael, Bradley, John, Sakimura Nat, RFC 7519: JSON Web Token(jwt), 2015

[8] Son, Jin-Kwang, "Vulnerability Analysis and Secure Coding of JWT-based Authentication Server", Proceeding of KISS 2016

[9] T. McLean. Critical vulnerabilities in JSON Web Token libraries, March 31, 2015

[10] TOTP: Time-vased One-time Password Algorithm, https://tools.ietf.org/html/draft-mraihi-totp-timebased-00.

[11] RFC 4226,HOTP: An HMAC-Based One-Time Password Algorithm, http://tools.ietf.org/html/rfc4226.