Blockchain-Based Two-Factor Authentication for Credit Card Validation

Suat Mercan[1(✉)], Mumin Cebe[2], Kemal Akkaya[1], and Julian Zuluaga[1]

[1] Florida International University, Miami, FL 33199, USA
{smercan,kakkaya,andrew}@fiu.edu
[2] Marquette Univesity, Milwaukee, WI 53233, USA
mumin.cebe@marquette.edu

Abstract. The widespread adoption of the e-commerce and web-based business has brought great increase in credit card utilization for online transactions which in turn resulted in sophisticated fraud attempts. Accurate fraud prevention and detection is a key concern in cashless economy. Multifactor authentication among others such as machine learning based behavioral analysis, data mining, black listing is one of the effective methods augmenting primary information checking. SMS messages are sent to registered phone in addition to credit card information as a second level protection. However, this information might be vulnerable to various attacks as some third party services are in the game. This paper proposes adoption of blockchain as a secure platform to store the second factor security information. User's mobile device signature attested by the bank is stored in a permissioned blockchain. This information is accessed by the merchant through user-friendly QR-code reading interface in order to verify that the user has the registered device. We present system design along with potential threats and security analysis.

Keywords: Blockchain · Two-factor authentication · Fraud detection

1 Introduction

Online credit card transaction is the base of e-commerce where anyone can purchase goods and services from merchants across the world. Unfortunately, payment cards are susceptible to fraud, and the majority of card fraud happens during online payments [6]. Deep consumer adoption of the technology has made credit card companies struggle improving and securing the technology against increasingly complex and evolving fraudster schemes. Credit card fraud has been a standing issue for financial institutions. Worldwide fraud loss is around 25 billion dollars and Card Not Present (CNP) accounts for more than 70% [1].

Current fraud reduction methods appear in various forms. Card verification value (CVV2) and Address verification Service (AVS) are the basic methods [6]. Fraudsters can easily obfuscate their data to defeat the conventional detection methods that rely on device and browser fingerprint information such as

J. Garcia-Alfaro et al. (Eds.): DPM 2021/CBT 2021, LNCS 13140, pp. 319–327, 2022.
https://doi.org/10.1007/978-3-030-93944-1_22

IP address, geolocation etc. In addition to this information, cardholder's spending behaviour is also analysed to determine the legitimacy using statistical and machine learning based methods [4] which is in fact a binary classification problem. Transactions are monitored based on certain parameters such as user's location, amount, purchase category etc. Despite the continuing efforts, current methods are still susceptible to false positive and false negative decisions. 97% of flagged transactions are actually legitimate which causes inconvenience and unsatisfactory shopping experience [2] and decreased shopping rate.

Such frauds especially put e-commerce merchants into a difficult situation. This is because, while card present (CP) losses are beared by the banks, merchants are still held accountable for CNP [10]. Because of this liability burden, e-commerce merchants have great incentive to prevent or reduce card fraud actively. In this respect, one of the solutions adopted by them is two-factor authentication (2FA). Most common 2FA method is sending one time password via an SMS message to a registered phone number. However, this technique is also vulnerable to various attacks such as man-in-the-middle (MitM) as the data goes through third party service provider.

To this end, in this paper, we propose a blockchain-based out-of-band verification scheme which relies on the cooperation of merchant and issuer bank. The proposed system is a second layer security feature that can be incorporated if desired by merchant with cooperation of banks. We consider e-commerce merchants to combat online transaction fraud effectively. The proposed system requires the bank storing a pre-approved signature of user device with its attestation in the blockchain. Then, the merchant is able to control the device signature on the blockchain to check if they match. After this first authentication, the merchant goes to second regular check via payment platform such as Visa. The proposed approach is evaluated by determining potential security threats and how they are addressed in the framework. In addition, we performed some tests on hyperledger to evaluate the feasibility.

The reminder of this paper is organized as follows: Sect. 2 presents preliminaries. In Sect. 3, we explain the proposed approach. Section 4 presents evaluation of the approach.

2 Related Work

Various fraud detection methods have been introduced including multifactor authentication, machine learning based methods [3]. CVV, 3 digit security number, and AVS are fundamental information required from user. Behavioral analytics and fingerprinting refers to monitoring suspicious activities and pattern for which supervised and unsupervised methods are utilized to model Some researchers label this problem as an outlier detection or anomaly detection problem [9]. Anomaly detection is looking into key factors, such as IP addresses, to detect anomalies in transactions. However, fingerprinting can be hidden by fraudsters. Supervised fraud detection methods rely on a set of previously known and labeled fraudulent transactions. Once similar transactions are identified by the

model that have certain attributes, they are tagged and classified as fraud and the system can decide not to process it. These models can expand to learn fraud based on tagged transaction that are later classified by an administrator. Methods based on Bayesian Network, Decision tree, Support vector machine exist to tackle this problem. While the machine learning algorithms may give false positives and negatives which either may result in fraud or customer inconvenience. Two factor authentication [7] has also been adopted in order to increase the level of security. The general approach is sending one-time password to the user via registered phone. Transmitting a text message through a third party may be susceptible to man-in-the-middle (MitM) attack.

3 Preliminaries

3.1 Background

For every online credit card transaction there are essential parties involved: *client cardholder, merchant, issuing bank, acquiring bank, a payment processor, and a credit card network* as shown in Fig. 1 [8].

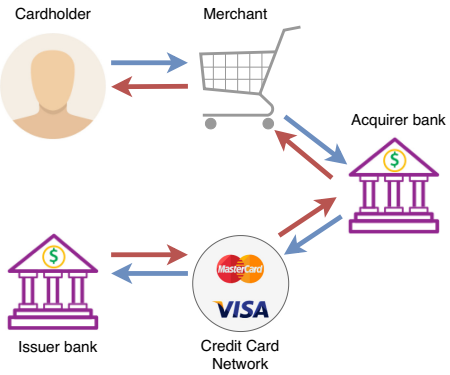

Fig. 1. Credit card payment system.

In an e-commerce CNP scenario, merchants need payment processors services in order for them to accept credit card transactions on their website. When a user decides to make a purchase online, the merchant will require a form to be filled out with pertinent details regarding the payment method. The payment processor will transfer the information to the acquiring bank, the bank will forward to the credit card network, who checks that the issuing bank's account has the funds available. The credit card network requests the payment authorization from the issuing bank including details from the card: credit card number, card expiration date, billing address for AVS, CVV, and payment amount [8]. The issuing bank receives the request from the network and validates that all the details are correct and that the funds are available. Then it will approve the transaction and send back an approval code through the credit card network.

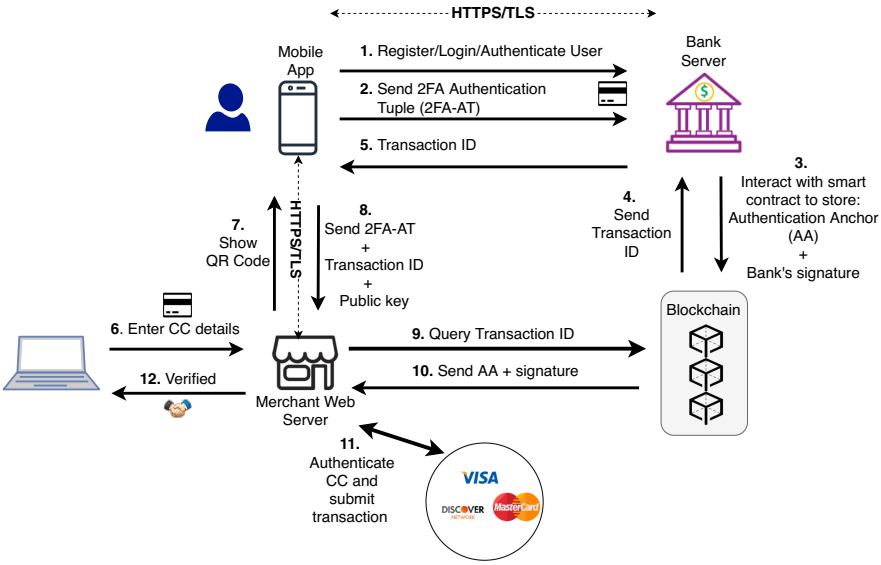

Fig. 2. Overview of 2FA platform for secure payments.

3.2 Attack Model

In this paper, we consider a credit card fraud scenario, which is very common in U.S using stolen credit cards' information. The adversary is determined to be a professional systematic fraudster who is in possession of many stolen credit card details with lucrative purposes. The security of the blockchain-based authentication framework depends on the secure implementation of proposed 2-FA system. Therefore, we consider the following threats to the security of the proposed approach and identified the relevant security goals. Note that in our attack model, we assume that the mobile device is tamper-proof through Hardware Security Modules (HSMs) that provide device-level controls to protect deployed keys. Therefore, mobile device infiltration is out of scope. In addition, our model assumes that merchant and bank are trusted parties.

Threat 1: In this attack, the attacker disguises itself as a mobile device user for pushing false authentication attestation into our multi-factor framework by obtaining the private keys that are used to sign the data.

Threat 2: In this attack scenario, the attacker attacks the mobile device communication layer and performs a man-in-the-middle (MitM) attack for altering the transactions.

Threat 3: In this attack scenario, the attacker can counterfeit data in the permissioned Blockchain.

4 Blokchain-Based 2-FA Framework for Transaction Authentication

4.1 Overview

The overall framework can be described as a blockchain-based 2FA approach consisting of smart contracts running on a permissioned blockchain along with some off-chain operations that bridge the end-users and banks as shown in Fig. 2. The permissioned blockchain will just run among banks as a result of a consortium agreement between them.

The first phase of the proposed framework is the registration that starts with generating the *2FA authentication tuple* (AT) by the end user's mobile device. The *2FA authentication tuple* is a bundle of device ID, a random challenge, and public-private key pair that acts as a unique, persistent authentication factor. The user sends the tuple to securely completing the registration of the mobile phone as a 2FA device. The bank combines the tuple and CC information to produce an *authentication anchor* (AA) by hashing the combination. Then, the bank submits produced anchor to blokchain by interacting with a smart contract on the blockchain.

When purchasing an item, the customer uses the merchant website and provides card-holder information such as card number, CVV, expiration date, etc. The merchant then verifies that who has provided the card information also has the mobile device associated with that card by interacting with blockchain and the mobile device. After this step, the merchant is now ready to initiate a transaction over the regular payment systems (e.g., Visa, Mastercard, etc.). With these steps, the merchant first ensures that the customer who provides CC information also carries the associated mobile device by interacting with the merchant and the card is still valid with a sufficient balance.

4.2 Registration Phase

The system first requires registration of the end user's device to the system. To do so, the end-user logins to the bank using its regular credentials over the bank application ❶. Then, s/he performs registration by sending the 2FA tuple, which is a bundle of the device ID (e.g., IMEI, Unique AppID, etc.) and a nonce. In addition to that, user device generates a public-private key pair for this device registration and sign the hash of the bundle. The 2FA tuple contains hash of the device ID, random challenge bundle and signature of the hash by using the freshly generated private and public key couple ❷. The bank then creates an AA by a combination of 2FA and hash of CC values with its signature. The bank triggers a smart contract called "Registration Contract" to cryptographically bind trusted bank identity and an off-chain 2FAtuple with CC information on the blockchain ❸. The Registration Contract acts as a logically centralized but physically decentralized lookup table mapping each bank, CC, and mobile device IDs. When the blockchain transaction is confirmed, the bank relays the transaction ID of the registration step to the mobile device, which will

be used in the system again ❹, ❺. When the user wants to update the AA, it just creates another registration phase following the same process. But this time bank first burns the previous anchor of the user and stores the new one on the blockchain by triggering "Burn Contract" which basically re-signs the previous AA with a public address known as "eater address". This is viewable by all nodes and invalidates that AA. The status of these coins is published on the blockchain.

This framework allows the user to view the bank as a proxy while interacting with smart contracts and introduces a secure layer between the user's mobile device and the blockchain while transferring the AA. This structure also helps a convenient way to update the user's mobile device while maintaining a persistent AA. In cases where the user's mobile device was able to submit its AA to the blockchain directly, the user would lose control over the previous AA when the device that holds the previous AA was lost. However, with this arrangement, the bank works as a delegation point for the user to recover its 2FA anchor and connects it to a new device.

4.3 Second Factor Authentication Phase

The second-factor authentication defines how the end-user interacts with the merchant website to perform a secure transaction and ensures that only 2FA authenticated card information is used for purchases. This step's primary goal is to manage the end user's registered device information to perform the requested purchase. The main idea is that the merchant has to authenticate the CC information via interacting with the registered mobile device before issuing the purchase order. In addition, quality of user experience is one of the preeminent goals of this phase while interacting with merchants, blockchain, and end-device. Thus, the interaction between merchants and end-users is established through a *QR code* mechanism. This provides a usable environment for a user to submit the required AA to the merchant to confirm that the CC belongs to him. In fact, the AAs are technically *zero-knowledge proofs*, which means that the phone-owner can prove his signature without sharing his information.

When a user presents CC information to carry out a purchase ❻, the merchant performs a 2FA before forwarding the transaction to existing payment systems. In order to do that, the merchant generates a QR code with a fresh random challenge to prevent a replay attack. The QR code contains the challenge, provided CC, and Rest API URL Address to retrieve required proofs from the user-end device ❼ using a challenge-response protocol. The user scans this code with the app and is presented a verification screen where s/he can verify the interaction using his/her fingerprint. Once the action is taken, the mobile app returns signed challenge data with the associated private key, signed 2FA tuple, transaction ID of blockchain, and corresponding public key ❽.

The merchant queries the blockchain via "Query Contract" by providing the transaction ID (e.g., retrieved from the mobile device) and fetches the stored AA to confirm CC and mobile device validity ❾, ❿. First, it checks whether the

hash of CC equals to the first of AA, which contains the hash of CC information. Then, it checks whether it confirms the signature of 2FA tuple with the second part of AA by using the fetched public key from the mobile device. Finally, it also confirms the signature of challenge-response message using the same public key. These steps ensure that this CC and 2FA tuple are associated before in an immutable way. The user is able to confirm signatures of both stored AA on blockchain and fresh challenge-response.

After these steps, the merchant is now certain that the user who provided CC details also carries the registered device. Finally, it forwards the CC details to the payment system to confirm that it has enough balance and is still valid (11). If the payment system confirms the transaction, the merchant and user successfully complete the process (12).

5 Evaluation

In this section, we evaluate the proposed framework in terms of its security features and performance via implementation.

5.1 Security Analysis

In this section, we consider all the attacks mentioned in our Threat Model in Sect. 3 and analyze how our proposed framework addresses these attacks.

Threat 1: In this scenario, the attacker tries to masquerade a mobile device for submitting bogus AA into the blockchain system. To do so, the attacker needs to derive the private keys of the mobile device to prepare that attestation. We argue that even if the attacker may access to a mobile device by leveraging a vulnerability, the attack will be thwarted due to secure integrated chips (e.g., Titan-M and TrustZone) in modern mobile devices.

Threat 2: In this attack scenario, the attacker may perform a MitM attack on communication channels between the mobile device and the bank, the bank and the blockchain peers, the merchant and the mobile device, the merchant and the blockchain peers. As described in Sect. 4, these communication channels are protected by public-private key pairs of parties by creating a secure tunnel and thus prevents any modification attempts. It is also important to note that our system addresses the threats coming from mobile operator in regular SMS-based 2FA systems. Since our system eliminates mobile operator, thus removes the risk emerging from this point.

Threat 3: In our framework, blockchain acts as an unbreakable seal to provide the integrity of the submitted AA. This is due the fact that, permissioned blockchain is a network that contains many parties. This makes our platform very secure against any single point of failure attacks since the validation of an AA depends on many validators. Thus, modifying a transaction requires fooling all parties to confirm that update, which is not possible.

5.2 Hyperledger Performance

We also implemented a Proof-of-Concept adapting Hyperledger Fabric [5] to test the feasibility of our framework in terms of latency and throughput. Hyperledger is a permissioned blockchain platform where access is restricted to stakeholders unlike the public blockchain where anyone can access the produced blocks. The blockchain consortium is expected to include participating banks where each entity should run a node. The main reason using a hyperledger like permissioned blockchain is that only participating banks should have access to the platform.

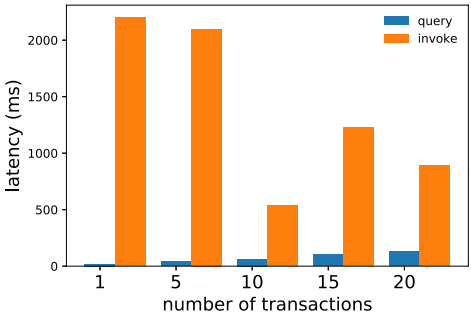

Fig. 3. Blockchain transaction times.

Since the time is crucial for customer satisfaction and transaction delay is the dominant factor in our system, we looked in to transaction latency for write (invoke) and read (query) operations. A bunch of transactions, from 1 to 20, is submitted to Hyperledger concurrently in order to test the performance. The results are shown in Fig. 3. As seen, the query latency increases from 20 ms to 100 ms with increasing number of transactions, which still can be considered fast. Average time of invoke operation, on the other hand, depends on the number of transactions submitted together and the blocksize. For a single transaction, it takes around 2 s because it waits for block timeout to submit the block for validation. When we look at 10 transactions, the average latency is much lower because the block is submitted immediately. This pattern will continue up to a saturation point which is generally measured as 140 tps, then it will start increasing.

6 Conclusion

In this paper, we proposed a CC fraud prevention system that preauthorizes online transactions using an out-of-bound authentication method, ultimately as an effort to protect merchants from fraud. The system utilizes a permissioned blockchain framework to store user information attested by the bank. The merchant is able to ask the purchaser to prove this transaction as authorized by

scanning a unique QR code with the mobile app. The merchant can verify the user by checking the blockchain. We provided the security analysis and performance evaluation to demonstrate our framework's security and feasibility.

References

1. Us payments forum. Card-not-present fraud around the world. https://www.uspaymentsforum.org/wp-content/uploads/2017/03/CNP-Fraud-Around-the-World-WP-FINAL-Mar-2017.pdf
2. Verifi inc. what every card not present merchant should know. https://www.verifi.com/wp-content/uploads/2014/05/Verifi_eBook_web_noCNP.pdf
3. Adepoju, O., Wosowei, J., Jaiman, H., et al.: Comparative evaluation of credit card fraud detection using machine learning techniques. In: 2019 Global Conference for Advancement in Technology (GCAT), pp. 1–6. IEEE (2019)
4. Awoyemi, J.O., Adetunmbi, A.O., Oluwadare, S.A.: Credit card fraud detection using machine learning techniques: a comparative analysis. In: 2017 International Conference on Computing Networking and Informatics (ICCNI), pp. 1–9. IEEE (2017)
5. Cachin, C., et al.: Architecture of the hyperledger blockchain fabric. In: Workshop on Distributed Cryptocurrencies and Consensus Ledgers, vol. 310, p. 4 (2016)
6. Conroy, J.: Card not present (CNP) fraud in a post-EMV environment. Aite Group, June 2014
7. Deshe, W., Chen, B., Chen, J.: Credit card fraud detection strategies with consumer incentives (2018)
8. Papadimitriou, O.: How credit card transaction processing works: steps, fees & participants. Wallethub, Abril (2009)
9. Porwal, U., Mukund, S.: Credit card fraud detection in e-commerce: an outlier detection approach. arXiv preprint arXiv:1811.02196 (2018)
10. Roggio, A.: 3 reminders about online payment fraud (2018)