Digitizing Invoice and Managing VAT Payment Using Blockchain Smart Contract

Van-Cam NGUYEN[§], Hoai-Luan PHAM[‡], Thi-Hong TRAN[‡], Huu-Thuan HUYNH[§], Yasuhiko NAKASHIMA[‡]
[§] University of Science, Vietnam National University, Ho Chi Minh City, Vietnam.
[‡]Graduate School of Information Science, Nara Institute of Science and Technology (NAIST), Ikoma, Nara, Japan.
Email: vancam1995@gmail.com, hhthuan@hcmus.edu.vn, {pham.hoai_luan.ox7, hong, nakashim}@is.naist.jp

***Abstract*—Nowadays, the authenticating procedure for transactions is still complicated, and the current Value-Added Tax (VAT) administration system works as centralized server, which consists of high-risk attacks from hackers. Therefore, only a few countries use digital technologies to calculate and manage the VAT payment despite of their benefits. In this paper, by combining decentralized storage network (DSN) with the smart contract (SC), we propose a new model based on Blockchain technology to authenticate the transaction, calculate Value-Added Tax and approve VAT payment. This system runs in a host computer (host PC) for encrypting and decrypting data. The smart contract is implemented in Remix Integrated Development Environment (IDE) based on the Ethereum platform. Experiential results show that the new model not only saves the cost for authenticating transactions but also protects the data from hacker's attacks due to consensus property of the Blockchain technology.**

***Index Terms*—Value-Added Tax, electronic invoice, Blockchain, smart contract, Ethereum.**

I. Introduction

In recent years, with the rapid development of the technology, enterprises (such as company, bank, restaurant, store, etc) are gradually digitizing all business documents. These save more storage space and processing cost than paper data. The electronic invoice significantly reduces exchange costs and the transactions become more convenient. According to allure surveys by Ernst & Young [1], 52% reviewers said that the key benefit of electronic invoice (e-invoice) is low cost. The cost to exchange each paper invoice is €7, while in electronic format it charges only €0.3, which reduces the cost by 25 times. In addition, each person can process only 6,000 paper invoices in a year, meantime a person can issue up to 90,000 invoices with the electronic form. E-invoices will be efficiently combined with the automatic digital system, which treats the e-invoice as input data. Although the e-invoice system brings many benefits, this paradigm is still complex. For instance, the Digital Invoice Customs Exchange (DICE) paradigm [2], which has been applied in the countries belong to the European Union. This paradigm needs totally four electronic signatures (one electronic signature's seller, one electronic signature's buyer, and two electronic signatures from the administrator) for a legal invoice. In order to get a lawful invoice in this model, the businesses have to operate through eight exchange steps (among seller, buyer and administrator). Apparently, the DICE paradigm is complex because of encrypting and decrypting invoice at each step. Moreover, the third parties (i.e., administrator) are required to authenticate transaction in the system. These problems lead to a high cost for transferring invoice to authenticate, encrypt or decrypt invoice at each party, and the third party rentals.

Authenticating invoice is the first step for VAT calculation. The lawful invoice will then be saved in the specific format (e.g., .PDF, .XML), which is used to determine VAT. As an example, one typical VAT administration model is the Standard Audit File for Tax (SAF-T). This model has been applied at the countries belonging to the Organization for Economic Co-operation and Development (OECD) [3] (e.g., Portugal, United Kingdom, Denmark, Belgium, Italy). The system audits tax (includes VAT) based on *.XML* format files. Via web services (HTTP or HTTPS protocol), these *.XML* files have been delivered to administrators by taxpayers. The tax administrator is SAF-T servers that store all legal invoice. There is a high risk in the centralized storage system. The data in servers may be able to be changed or destroyed by hackers unexpectedly, even though the information of invoice had been encrypted. Furthermore, if there is a huge number of user request to the server at the same time, server will be overload, which slows down the speed to access data (e.g., DDoS attacks).

To address the issues mentioned above, we propose the full VAT administration system based on Blockchain smart contract. Contributions of this paper are as follows:

Firstly, we propose a new model to authenticate invoice by using Blockchain smart contract. All information of invoice is encrypted and stored in the Decentralized Storage Network (DSN) through InterPlanetary File System (IPFS) protocol. The hash code, which is returned from DSN, will be uploaded to the Blockchain network (BCN). The hash code presents to encrypted invoice. This invoice will only be authenticated by sellers and buyers through the smart contract (SC) without the third party (VAT administrator). Our proposed model operated in three steps to get the lawful invoice instead of eight steps in the DICE paradigm.

Secondly, based on the legal invoice, the smart contract will calculate VAT payment. All information of VAT calculation and payment is stored in the BCN. This data is hardly attacked to change by hacker because of consensus property

This work was partly supported by JSPS KAKENHI Grant Number JP16K18105 and Kinokyoka Collaboration Research Fund, NAIST.

978-1-7281-1328-9/19/$31.00 © 2019 IEEE

of Blockchain technology. Furthermore, all data is broadcasted BCN entirely, leads to speed up data access even if there are many requests for data synchronously.

Finally, the accuracy, cost and security of the new model were analyzed.

The remain of this paper is organized as follows. Section II shows the background. In Section III, we describe in detail the proposed system. We also present the implementation scenario and experimental result in Section IV. Finally, the conclusion is given in Section V.

II. Background

A. Blockchain technology

Blockchain technology operates as the distributed ledger structure, which stores all data by the time and broadcast them to entire network. It is nearly impossible to change the decentralized data, because the data is only reversed if there is more than 51% agreement from all nodes in the network [4][6][7][17].

B. Value-Added Tax (VAT)

The Value-Added Tax, or Goods and Services Tax (GST) in some countries, is a kind of tax, which is charged in case the price of goods or services increases at each stage of the production process and distribution [5] (e.g., from manufacturer to retailer to customer).
The periodic VAT payment (PVP) is calculated as formula (1).

$$PVP = [\sum_{i=1}^{n}(V_i) - \sum_{j=1}^{k}(E_j)] * r \quad (1)$$

where n and k are the number of selling invoice and buying invoice, respectively. The periodic payment for VAT is calculated by multiplying the subtraction between the total revenue V (includes VAT) and total the input expense E (includes VAT) with the corresponding VAT rate r. This rate depends on each country [5] and types of goods.

C. Smart contract and Ethereum platform

- **Smart contract** is version 2.0 of the Blockchain technology[16]. This is short machine program, which defines conditions of contract in the business. These short programs are executed without the third party automatically. Applying smart contracts increases the trust between parties [8][9].
- **Ethereum platform** operates based on Blockchain technology and executes scripts by using Ethereum Virtual Machine (EVM)[10]. In the Blockchain network, nodes are developed based on Ethereum platform. These nodes use a pair of key, Private key (PrK) and Externally Owned Account (EOA) to communicate each other [13].
- **Remix IDE** is an application, which provides the environment for scripting, compiling, debugging and executing the smart contract using Solidity language [11].

III. Proposed system

A. System model

The overview of our proposed VAT payment system is shown in Fig. 1. The system has four entities. They are VAT administrator (VAD), sellers, buyers and the bank. To reduce storage space in the BCN, we encrypt invoices and store them to DSN via IPFS protocol. Then, the DSN will return the hash code, which presents the location of data in DSN. This hash code will be uploaded to the BCN after that. The operation of system can be explained via steps as follows:

① VAD deploys the smart contract into BCN.
② VAD adds the list of registered VAT payer to the smart contract.
③ The seller (included in the list of registered VAT payer) stores encrypted invoices to DSN.
④ The seller records hash code.
⑤ The seller uploads information of transaction (e.g., hash code,value of goods, status) into BCN.
Value of goods helps the smart contract calculating VAT efficiently without decryption. Although all users in BCN can get this value, users cannot detect the owner in reality business because of anonymous property as shown in II.A section.
The status is used to recognize the buyer whether is the end customer or not. If the buyer is not the end customer, the lawful invoice is created if there are enough two verifications from seller and buyer. By contrast, only seller's verification is enough.
⑥ The buyer gets hash code from BCN.
⑦ The buyer decrypts the encrypted invoices with corresponding hash code in DSN.
⑧ The buyer verifies the invoice into smart contract.
⑨ The seller verifies the invoice after buyer's verification. The legal invoice is created immediately. Then, the smart contract automatically calculate VAT for corresponding payers.
⑩ In periodically, VAD requests for periodic VAT payment. The smart contract assembles all VAT from invoices for corresponding payers.
⑪ VAD approves the periodic VAT payment. If there is any nonsense, VAD can trace back the information of transaction from DSN.
⑫ Based on the approval for periodic VAT payment from VAD, the bank disburses flow of money for VAT payment from the corresponding payer bank account to VAT administrator bank account.

B. Encryption and decryption

To protect invoice data, encryption and decryption are necessary. An asymmetric cryptography algorithm, Rivest–Shamir–Adleman (RSA) [14], is used in this new model. A pair of key is applied to encrypt and decrypt data. The plaintext *PT* is encrypted based on the public key (PUK) and the private key (PRK) is used to decrypted the ciphertext

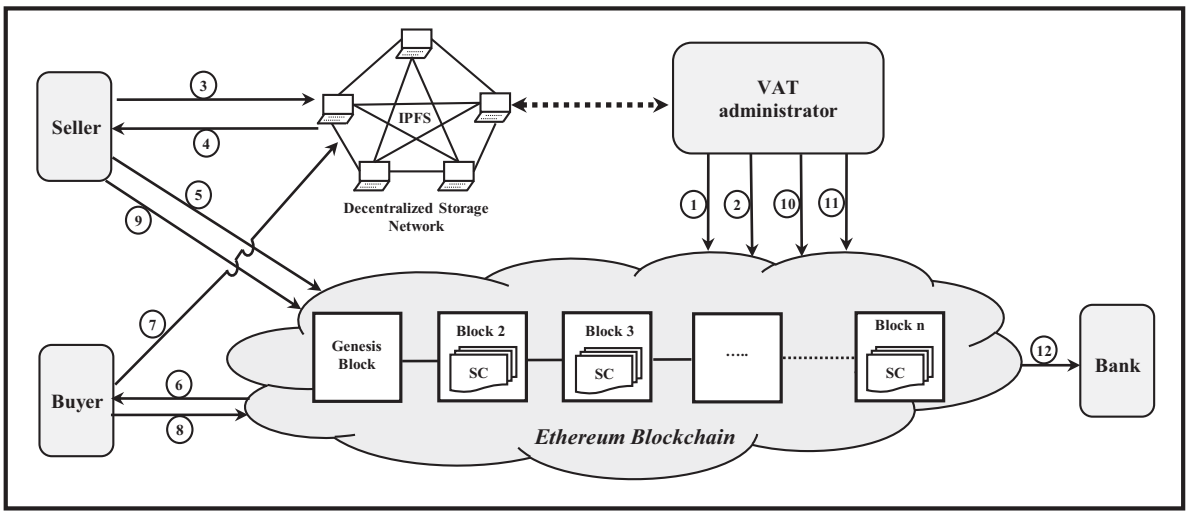

Fig. 1. The VAT administration system model based on Blockchain smart contract.

CT. In this model, buyers broadcast PUK to concerned parties (partner) and sellers keep PRK.

C. Smart contract design

Eight main functions are used in the smart contract for the proposed model. They are described as follows:
addVATPayer(): Only VAD, who deployed the smart contract can execute this function. VAD uploads the address of registered VAT payer.
addInforInvoice(): This function can be only run by sellers. Although the input includes buyer's address and price of invoice, the other users in BCN cannot detect who is the owner and which transaction the price belongs to.
agreeFromBuyer(): Only buyers can execute this function to verify the invoice.
agreeFromSeller(): Only sellers can execute to verify the invoice after buyer's verification. The lawful invoice will be created after seller's verification. Immediately, the smart contract will calculate the VAT for only seller if the buyer is the end customer. By contrast, VAT for seller and buyer is calculated.
requestForPVP(): In periodically (e.g., monthly, quarterly), VAD executes this function to request for PVP. Only VAD can run this function. The smart contract will assemble the PVP for the corresponding payer as formula (1).
agreePVP() and **disagreePVP()**: can only be executed by VAD. These functions are used to approve the final PVP for all transaction in periodically.
The bank runs ***getInforPVP()*** function to get status of PVP. The bank will disburse flow of money for VAT payment from the corresponding payer bank account to VAT administrator bank account.

IV. Implementation and Experimental result

A. Implementation

In this section, we mainly describe the implementation of our model. The scenario is shown in Fig. 2.
Firstly, a computer (host PC) is used to encrypt and decrypt files, which contains information of invoice. We run with a seller and a buyer in the same host PC with configuration as: Inter(R) Xeon(R) CPU E3-1275 v6 @3.80GHz, 16.0 GB physical memory (RAM). And, host PC runs in Microsoft Windows 10 Pro operation system.
We randomly generate a pair of key (PUK_Buyer and PRK_Buyer). The *Plaintext.txt* file is encrypted by using NodeRSA library based on JavaScript language. After that, *Ciphertext.txt* file is created and uploaded to DSN by using IPFS platform. Immediately, host PC records hash code, which is returned from DSN. By contrast in buyer's side (in the same host PC), we decrypt *Ciphertext.txt* file, which get from DSN with the corresponding hash code, is same as encrypt processes.
Secondly,we have used four account addresses that present seller, buyer, VAD and the bank, respectively. Sequentially, we have implemented same as steps in Fig. 1 in the JavaScript Virtual Machine environment through Remix IDE [15]. The smart contract has been written by Solidity language.

B. Experimental result

We analyze the cost for executing functions and present the accuracy in this section.
Table I shows the cost to execute the smart contract functions. As examples, $0.00587 is the cost to deploy an address of the registered VAT payer, $0.01974 is the amount paid for uploading information of an invoice, etc. According to table I, *addInforInvoice()* is the most expensive procedure in our model. In generally, the cost of operation in the proposed smart

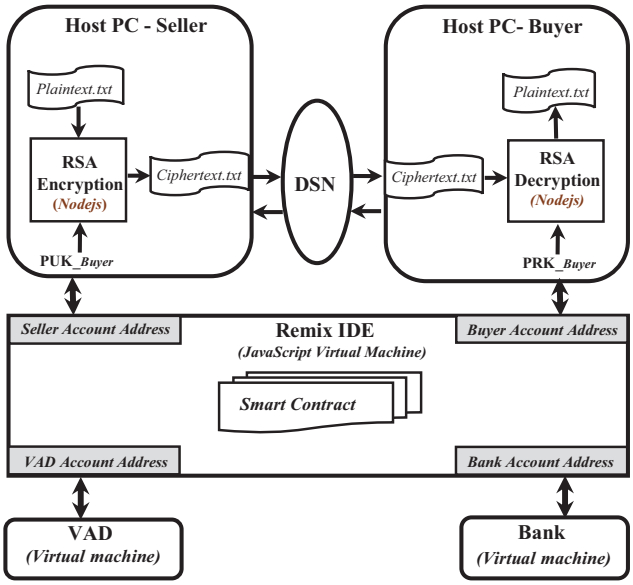

Fig. 2. The implementation model.

TABLE I
GAS COST OF FUNCTIONS IN THE SMART CONTRACT
(GAS PRICE = 2.4 GWEI, 1 ETH ≃ $87)

Algorithm	Gas used	ETH	USD
addVATPayer	41055	0.0000985	0.00857
addInforInvoice	94560	0.0002269	0.01974
agreeFromBuyer	28603	0.0000686	0.00597
agreeFromSeller	24022	0.0000577	0.00502
requestForPVP	31528	0.0000757	0.00659
agreePVP	20879	0.0000501	0.00436
disagreePVP	16528	0.0000397	0.00345
getInforPVP	0	0.0	0.0

contract is low. We accessed to ETH-USD market average price at 13:56 (GMT +9) on December 16th, 2018 [12].
The authentication property has been tested in our implementation. More detail, only VAD can execute *addVATPayer*, *requestForPVP*, *agreePVP* and *disagreePVP* functions. Moreover, only sellers can execute *addInforInvoice* and *agreeFromSeller* function, as long as *agreeFromBuyer* function can only be run by buyers. The result has been same as description in Section III.

C. System analysis

In the new model, we have combined decentralized storage network with Blockchain smart contract. This work brings many benefits for the VAT administration systems. The parallel discussion is given in this section.
Trust: In our model, based on the smart contract, the legal invoice is created and the VAT amount is calculated automatically without the third party. This cuts down cost to third party rentals. Moreover, our model reduces biased affection for verifying and VAT calculating process.
Availability: In the current VAT administration system, the document that concerns to VAT (i.e., invoice, VAT payment), is stored to centralized server. Instead, our system operates with decentralized model, which avoids being overload for accessing the data if there is a huge number of user request at the same time.
Immutability: The ability to change data is impossible nearly due to consensus property in the Blockchain technology. If there is any attacking to an user, the data will then be restored from the other users in the Blockchain network.

V. CONCLUSION

In this paper, we propose a system that digitizes invoice and automatically calculates VAT by using Blockchain smart contract. The smart contract has been implemented on the Remix IDE using Solidity language based on Ethereum platform. As the experiential results, the cost for digitizing invoice and calculating VAT is low in the new model. In addition, our proposed system decreases the risk of data loss attacks and improves the trust in implementing VAT payment (non-affection from the third party) as analysed.

REFERENCES

[1] Ernst & Young global organization,*"Worldwide electronic invoicing survey"*, pp. 4, 2018.
[2] Richard Ainsworth, Musaad Alwohaibi, *"Blockchain, Bitcoin, and VAT in the GCC: The Missing Trader Example"*, February 2017.
[3] Deloitte, *"Blockchain technology and its potential in taxes"*, December 2017.
[4] Giang-Truong Nguyen and Kyungbaek Kim, *"A Survey about Consensus Algorithms Used in Blockchain"*, vol.14, No.1, pp.101-128, February 2018.
[5] *"Taxation and Customs Union"*. [Online] Available at: https://ec.europa.eu/taxation_customs/business/vat/what-is-vat_en
[6] A. Beikverdi, J. S. Song, *"Trend of centralization in bitcoin's distributed network"*, IEEE/ACIS 16th International Conference on Software Engineering, 2015.
[7] C. Natoli, V. Gramoli, *"The blockchain anomaly"*, IEEE 15th International Symposium on Network Computing and Applications (NCA), 2016.
[8] G. Destefanis, M. Marchesi, M. Ortu, R. Tonelli, A. Bracciali, R. Hierons, *"Smart contracts vulnerabilities: a call for blockchain software engineering?"*, International Workshop on Blockchain Oriented Software Engineering (IWBOSE), 2018.
[9] L. Chen, L. Xu, N. Shah, Z. Gao, Y. Lu, W. Shi, *"Decentralized Execution of Smart Contracts: Agent Model Perspective and Its Implications"*, International Conference on Financial Cryptography and Data Security, Financial Cryptography and Data Security, pp 468-477, 2017.
[10] Hoai Luan Pham, Thi Hong Tran, Yasuhiko Nakashima, *"A secure remote healthcare system for hospital using Blockchain smart contract"*, IEEE Global Communications Conference, 2018.
[11] Remix documentation. [Online]. Available at: https://remix.readthedocs.io/en/latest/
[12] ETH/USD - Market average price, ethereumprice . [Online]. Available at: https://ethereumprice.org/
[13] Ethereum Development Tutorial. [Online]. Available at: https://github.com/ethereum/wiki/wiki/Ethereum-Development-Tutorial
[14] Hongwei Si, Youlin Cai, Zhimei Cheng, *"An Improved RSA Signature Algorithm Based on Complex Numeric Operation Function"*, International Conference on Challenges in Environmental Science and Computer Engineering, 2010.
[15] Remix IDE. [Online]. Available at: https://remix.ethereum.org/
[16] A. Reyna, C. Martín, J. Chen, E. Soler, M. Díaz, *"On blockchain and its integration with IoT. Challenges and opportunities"*, Future Generation Computer Systems, vol 88, pp 173-190, November 2018.
[17] N. Z. Aitzhan, et al, *"Security and privacy in decentralized energy trading through multi-signatures, blockchain and anonymous messaging streams"*, IEEE Trans. Depend. Sec. Comp., in press, 2017.