Peer-to-Peer Networking and Applications
https://doi.org/10.1007/s12083-020-01022-0

A secure vehicle theft detection framework using Blockchain and smart contract

Debashis Das [1] · Sourav Banerjee [2] · Utpal Biswas [1]

Received: 15 July 2020 / Accepted: 22 October 2020

Abstract
Nowadays, vehicle theft has been increased drastically all over the world. The prevention of vehicle theft is essential to enhance vehicle security. A large number of vehicles have been stolen for the lack of essential foundation and administration of a secure platform. Several existing systems were developed to protect the vehicle from unauthorized access. But, they suffer from various limitations such as data security, the possibility of cybercriminals activities, leakage of personal information, and centralized System. We propose a vehicle theft detection framework using a decentralized and secure platform to increase the security level of the vehicle. Blockchain and smart contracts are used to provide the security of the stored data on the ledger and authenticate a genuine user automatically with greater accuracy. In this paper, we present 2-Step Authentication (2SA) and unauthorized access detection algorithms. The 2SA ensures the secure accessibility of the application by providing the randomly token chosen by the user. The proposed framework can provide vehicle security and owner's privacy. In this proposed framework, more than one person can drive the vehicle authorizing by the vehicle owner without hampering stored data in the vehicle device.

Keywords Blockchain · Ethereum smart contract · Blockchain-enabled anti-theft system · Vehicle security · Vehicle theft detection

1 Introduction

For a long time, vehicles have a broad range of implications for social interactions. A vehicle can provide an efficient mode of transportation in our society. In recent years, vehicle theft has been increased globally. Many vehicle anti-theft systems were developed and implemented to reduce vehicle theft issues. But, it is still now a challenging task to reduce vehicle theft issues. Many vehicle anti-theft systems were developed using Internet of Things (IoT) [1], GPS (Global Positioning System) / GSM (Global System for Mobile Communications) technology [2], obsolete smartphones [3], fingerprint verification [2, 4], and face identification [5]. But, these systems suffer from data privacy and compromise with vehicle security. The existing methods also have potential risks from cybercrime as many communication systems are connected through the web service [6–8]. Security failures of these methods may contribute to the leakage of personal information, issue of security vulnerabilities in mobile applications, and make the possibility of data manipulation [9]. Meanwhile, user privacy and data security could be suffered without addressing these issues. Many existing systems also suffer for an adequate key management policy to identify the user as well as the vehicle. The above-mentioned systems were mainly monotonous system in case of vehicle driving. Multiple drivers cannot drive the vehicle without hampering the vehicle data.

An anti-theft system is required for the vehicle where a user can trust and utilize the system efficiently and securely. This efficient and secure system can be developed using the blockchain technology, which can provide a decentralized, secure, transparent, and trust environment [10]. Blockchain has immutable data storage. All the nodes in the blockchain are connected through the peer-to-peer (P2P) network [11]. Nowadays, blockchain has been implemented as an application in various domains such as healthcare management [12], supply chain management [13], smart appliances [14], cross

✉ Sourav Banerjee
mr.sourav.banerjee@ieee.org

Debashis Das
debashis2124@gmail.com

Utpal Biswas
utpal0172@gmail.com

[1] University of Kalyani, Kalyani 741235, India

[2] Kalyani Government Engineering College, Kalyani 741235, India

Published online: 06 November 2020

border payment [15], IoT applications [16], Internet of Vehicles (IoV) [17] and many more. So, vehicle security can be enhanced using the blockchain technology that can be utilized to provide a secure anti-theft system for vehicles.

In this paper, we have proposed a vehicle theft detection framework using blockchain and smart contracts. These are used to secure the stored information and to authenticate a user automatically (vehicle owner). This framework mainly comprises two parts, which are user authentication and unauthorized access detection of a vehicle. In the case of user authentication, the user is authenticated using the 2-Step Authentication (2SA). For the 2SA, a token-based approach is proposed to authenticate a genuine user of an account. A smart contract can easily detect an unauthorized person who tries to access another user account. Meanwhile, unauthorized access can be detected on a vehicle using the unauthorized access detection algorithm. Only the user can allow a person who wants to drive his vehicle. Thus, the proposed system can provide the proper security of the vehicle.

The main contributions of this research work are mentioned in the following:

- The 2SA algorithm is presented to authenticate a genuine user.
- An unauthorized access detection algorithm is proposed for protecting a vehicle from an unauthorized person.
- More than one person can be allowed by the user for driving the vehicle. The application of the proposed system automatically can update the vehicle's storage data.

The remaining part of this paper has been organized as follows: Section 2 describes existing vehicle anti-theft systems and their limitations. Section 3 explains the proposed framework and implementation procedure of the framework. Section 4 explains the experimental results. Section 5 provides a performance-based comparison analysis of the proposed framework with the existing system. Finally, section 6 states the conclusion and future direction.

2 Related work

There are few works on this nascent field of research i.e. Blockchain. There has no useful research work on the vehicle anti-theft system using blockchain. However, we have discussed some vehicle anti-theft systems that were developed using different methods.

Liu et al. [3] proposed a low-cost vehicle anti-theft system using the idea of PhoneInside by leveraging an extinct smartphone. In the proposed system, the smartphone can detect vehicle location and vehicle activities at the time of driving. The identification of vehicle theft can be detected using ad-hoc authentication. A timely alarming process and vehicle tracking can be performed using this system whenever it is required.

Dey et al. [2] proposed a GPS/GSM based anti-theft system for protecting vehicles. They used a fingerprint-enabled verification system to authenticate the driver. The system is implemented using low-cost technology with greater security.

In 2013, Zhigang et al. [1] presented a vehicle tracking system with the help of IoT. A mobile phone was used as a terminal for the development of the system. The system is used the RFID module to switch the control system and the GSM module to track the vehicle location. The use of IoT leaves their system vulnerable from cyber-attacks, where attackers can mangle confidential information.

Kiruthiga et al. [4] proposed a fingerprint-based system to protect vehicles from unauthorized access. The proposed system was built on an embedded platform using the PIC microcontroller. The vehicle device can send a notification to the owner of the vehicle using the GSM module. The system can provide the service even battery supply is stopped. Using GPS technology, the vehicle can also be traced easily in this system.

Liu and He [5] proposed an alert system for vehicles applying the face identification methodology. This system can trace an unauthorized person because of a facial recognition mismatch and send an image of the unauthorized person to the police station as well as to the vehicle owner. Once the vehicle starts moving, this device captures the video pictures of the driver using an activated IR illumination tool. Therefore it can detect the eyes and identifies the face of the driver applying the PCA algorithm.

Qian et al. [18] developed a tracking and alarming system for vehicles using an android-based operating system. Vehicles' details can be transmitted through a communication network to the owner's mobile phone. The system can send an alarm notification using a text messaging system.

Wei et al. [19] developed a low-cost distributed vehicles' wireless locating and monitoring framework. This framework is comprised of two components: a smartphone app terminal for a user and a vehicle terminal to capture vehicles' locations in real-time. But, communication-related threats may happen in the mobile network system.

3 Proposed vehicle theft detection framework

This section focuses on the proposed vehicle theft detection framework using blockchain technology [20] and smart contract [21]. The proposed solution utilizes the key features of blockchain and Ethereum smart contracts to achieve a decentralized, trusted, and secure vehicle theft detection system with the 2SA. The main features of the proposed solution including an unauthorized access detection on a vehicle, an

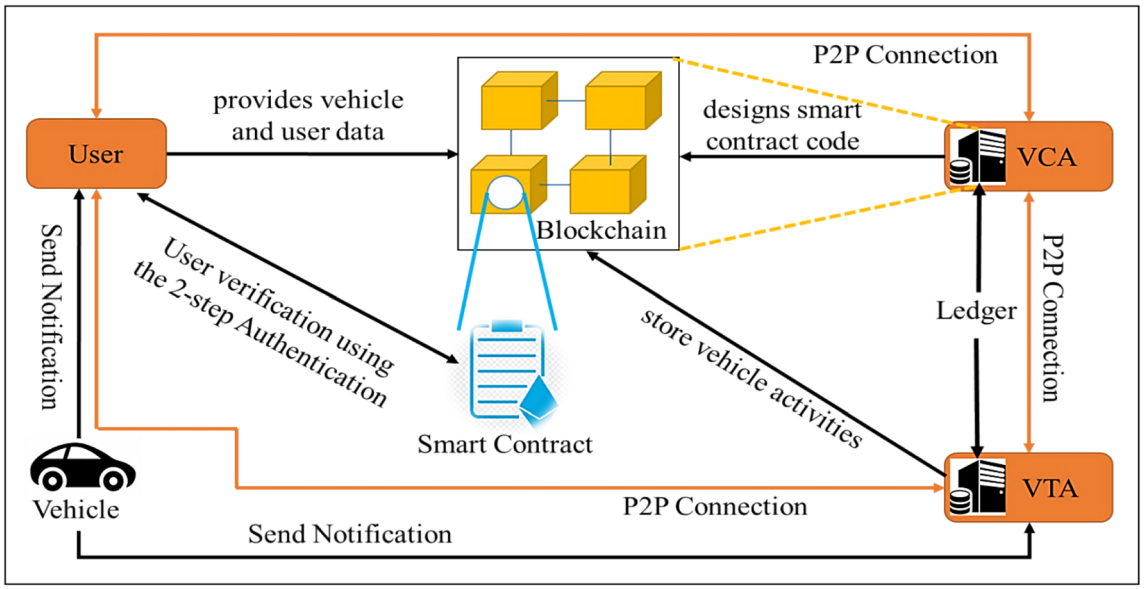


Fig. 1 System Overview of the proposed framework

Authorized Driver (AD) selection, current driver selection for driving, and deactivation of a driver.

3.1 System overview

The proposed solution focuses on the detection of unauthorized access to a vehicle. Here, all the connected entities through the peer-to-peer (P2P) connection in the blockchain network are User, Vehicle Certification Agency (VCA), and Vehicle Transportation Agency (VTA). The main role of each entity are described in the following:

- **User:** A user who has a vehicle, and can access the proposed framework's application by creating an account to the application site. After the creation of the account, the user provides the vehicle and personal details. All data provided by the user is stored in the application storage. The user can request any features through the user interface. The user doesn't need to keep a copy of the whole blockchain ledger. The user only utilizes the application for vehicle security.
- **Vehicle Certification Agency:** Vehicle Certification Agency (VCA) is an authenticated entity to mine a block.

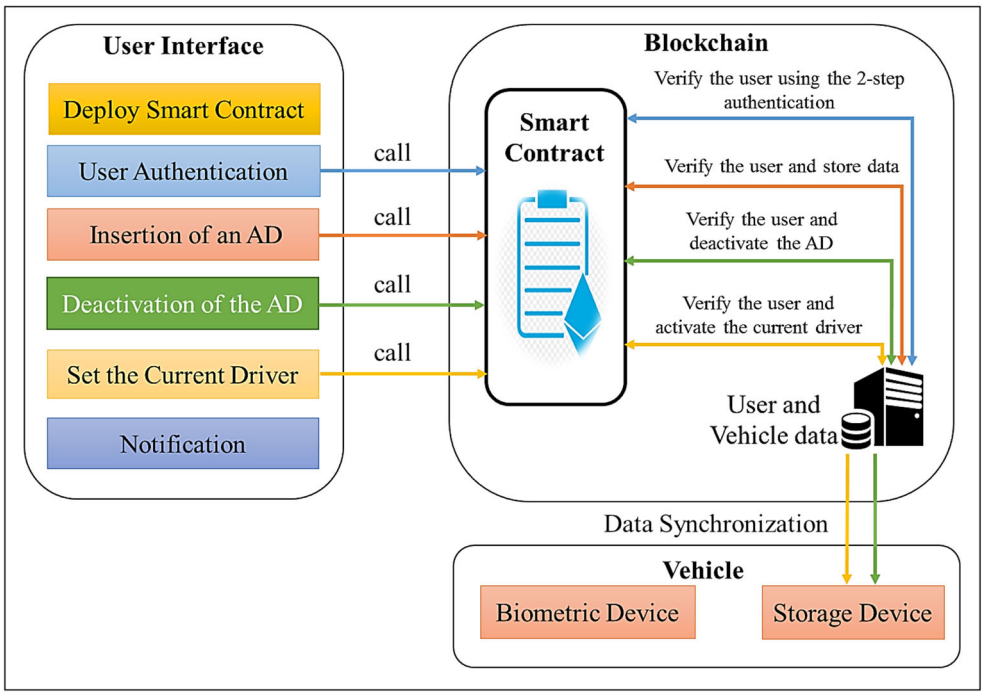


Fig. 2 Interaction between the user, the vehicle, and the smart contract

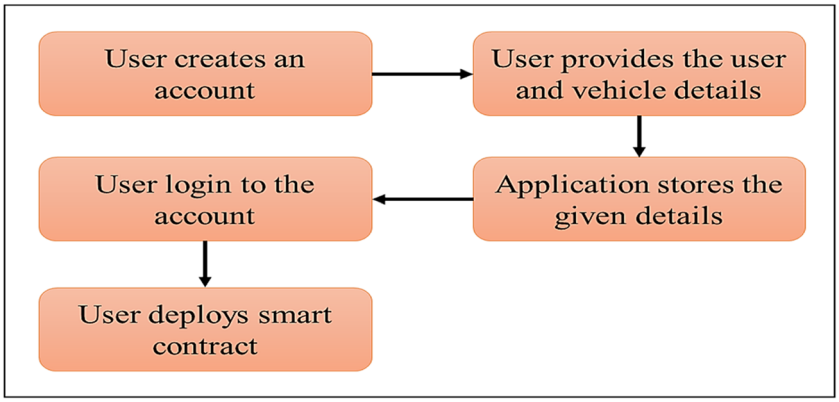


Fig. 3 Account creation and the smart contract deployment by the user

The VCA is also responsible for the design smart contract's code. The user can deploy this designed smart contract without having any knowledge of the code. It keeps a blockchain ledger to observe user activities and manage disputes handling.

- **Vehicle Transportation Agency:** Vehicle Transportation Agency (VTA) is responsible for storing vehicle activities and tracking the vehicle. The VTA receives notification from vehicle devices while unauthorized access has happened. As per the notification, the VTA can take a needful step. If the user cannot aware of his vehicle, the VTA will be tracking the vehicle until the vehicle status won't be received from the user. The status includes a safe or unsafe state of the vehicle which can be updated by the user. The VTA keeps a copy of the blockchain ledger to take immediate action if needed to identify and trace the vehicle.

- **Vehicle:** A vehicle has a small storage unit to store a driver data for the authorization. It can communicate with the application using the Control Area Network (CAN) [22]. It is a bus model of the vehicle, which is built to connect between microcontrollers and accessories in a device without a hosted machine. It is a message-based protocol, basically designed for multiplex electrical wires in a vehicle. Onboard Wireless Units (OBU) can communicate with the CAN-bus. In particular, it is a framework that transmits data through an OBU connector using a wireless Internet interface framework [22]. It is essential, simple, and particularly suitable to access and transmit the data.

The proposed framework is shown in Fig. 1. The user has to complete the 2SA for accessing features of the proposed framework. If the user has been authenticated successfully, the smart contract allows the user to access the vehicle. If the user cannot be authenticated, then the smart contract will reject all requests that have been received from the user. Therefore, a notification will send to the user and the VCA while the smart contract detects any unauthorized access on the user account. A notification also can be sent from the vehicle device to the user and the VTA while unauthorized access has been detected to the vehicle.

The smart contract has a significant role in this proposed framework, where it is deployed by the user to the blockchain. The user can request for available features through the user interface and the smart contract acts in the backend as per the user's request. Figure 2 shows that the user can request the smart contract through the application for accessing the available features such as *Deploy Smart Contract*, *User Authentication*, *Insertion of an AD*, *Set the Current Driver*, and *Deactivation of the AD*. However, the user must need to authenticate using the 2SA before requesting for the *Insertion of an AD*, *Set the Current Driver*, and *Deactivation of the AD*. We have discussed all the available features of the proposed framework in the following:

3.1.1 Account creation and smart contract deployment

Every user has an Ethereum account address. A user can sign up on the application website, and provide the required information and the Ethereum account address. The user can log in to the application using the email and password. Therefore, the user has to deploy a smart contract by clicking the *Deploy Smart Contract* function from the application site for accessing all the features. The user cannot access the smart contract's code. Figure 3 shows the process of the account creation and the smart contract deployment by the user.

3.1.2 2-step authentication

In this proposed framework, 2-Step Authentication (2SA) is provided to authenticate the user. In the 1st step authentication (1SA) process, the user has to login using the username and password. If the provided username and password are correct, then the user will be authenticated successfully for the 1SA only. Therefore, the user has to set a random token and provide this token to the application's smart contract. The smart contract stores this token using his Ethereum account address. Therefore, the application asks the user to provide the token.

Then the user sends it to the application's smart contract and the application renders the stored token. If the token is matched, then the user will be authenticated successfully. Therefore, the 2SA process is completed successfully. If one of the two steps of the authentication is failed, the user cannot access any features of the proposed system.

Figure 4 shows the complete process of the 2SA, and the 2SA algorithm has been provided in the following:

Algorithm 1: 2-Step Authentication (2SA)

Input: UserToken

1. **Start**
2. user account address has been already saved in the smart contract;
3. user login to the application;
4. flag=false; // flag indicates token is set or not;
5. **if** (login is valid)
   then
   a. 1st step authentication is completed successfully;
   b. Now, user is authenticated to set the token;
   c. user=msg.sender;
   end
   if (UserToken>0)
   then
   a. User provides the UserToken;
   b. StoreToken==UserToken; // the smart contract saves the UserToken given by the user to the StoreToken //
   c. flag=true; // its indicate that the user sets token successfully //
   end
   else
   invalid token;
   end
6. **else**
   invalid user
   end
7. **if** (msg.sender==user)
   then
   a. if(UserToken==StoreToken)then // User provides the UserToken to the smart contract for the 2nd step authentication //
   b. 2nd step authentication is successful;
   end
   else
   2nd step authentication is failed;
   end
8. **else**
   invalid user;
   end
9. **End**

3.2 Insertion of an authorized Driver's data

The user can insert an Authorized Driver's (AD) data to permit the AD for driving his vehicle. The AD is an authenticated person who can be allowed by the user to drive the vehicle of the user. The user can enter data through the user interface after completing the 2SA successfully. This feature is needed while a person wants to drive the vehicle. The provided

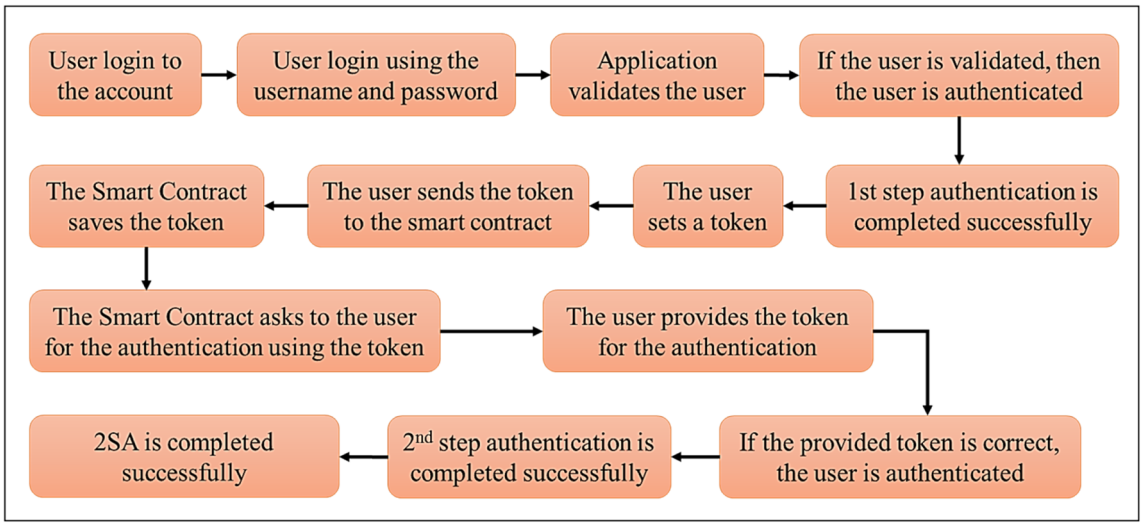


Fig. 4 Two-Step Authentication process

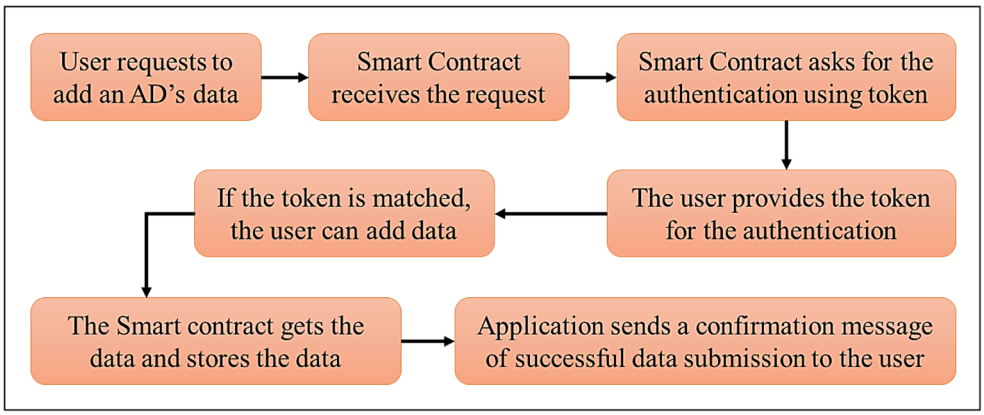


Fig. 5 Data insertion process for an AD

information of the AD will be stored in the application's smart contract, and a notification of successful data insertion will be sent to the user. The smart contract is a digital asset deployed to Ethereum Virtual Machine (EVM) that can store data [23]. We can use two types of variables in solidity [24]: state variables and local.

variables. These variables can store the state of the smart contract by storing the values in a block of the blockchain. State variables are storage by default, where information is stored in the blockchain [25]. The process of data insertion is shown in Fig. 5.

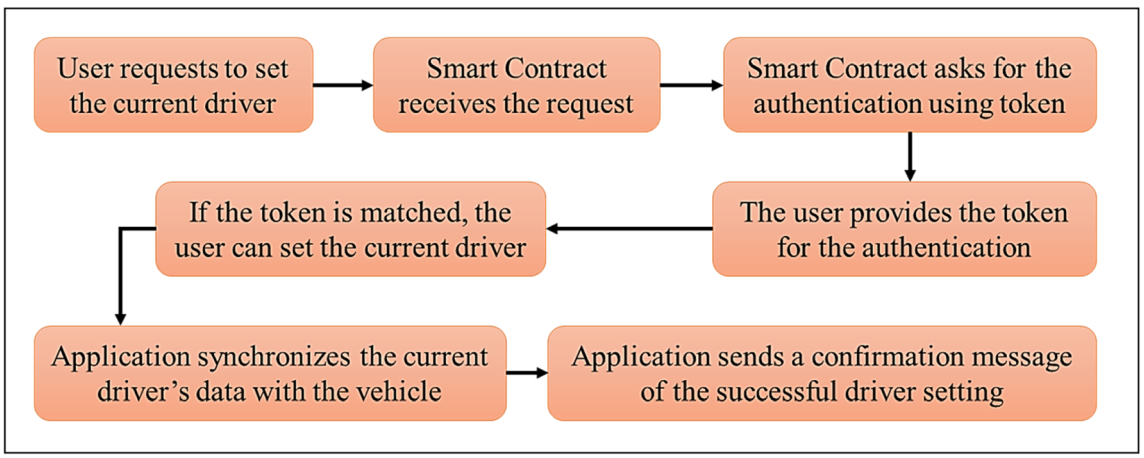


Fig. 6 The process of the current driver setting

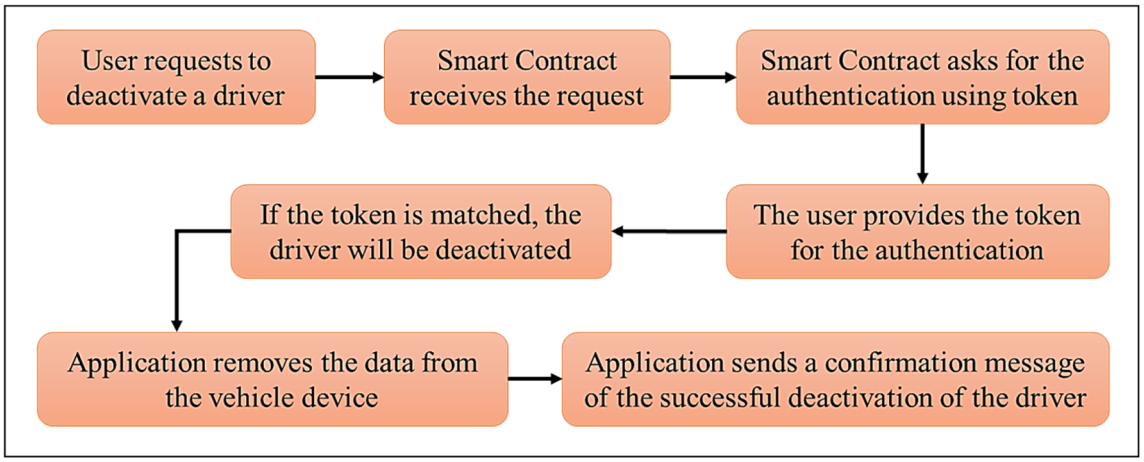


Fig. 7 The process of the deactivation of the AD

3.2.1 Set the current driver

The *Set the Current Driver* is an important feature of the proposed framework. The user can set the current driver who wants to drive the vehicle. The current driver can be the user himself or the AD. But, the user has to set the current driver before driving the vehicle. The user must need to complete the 2SA before setting the current driver. Therefore, the user can set the current driver. Figure 6 shows the process of the current driver setting.

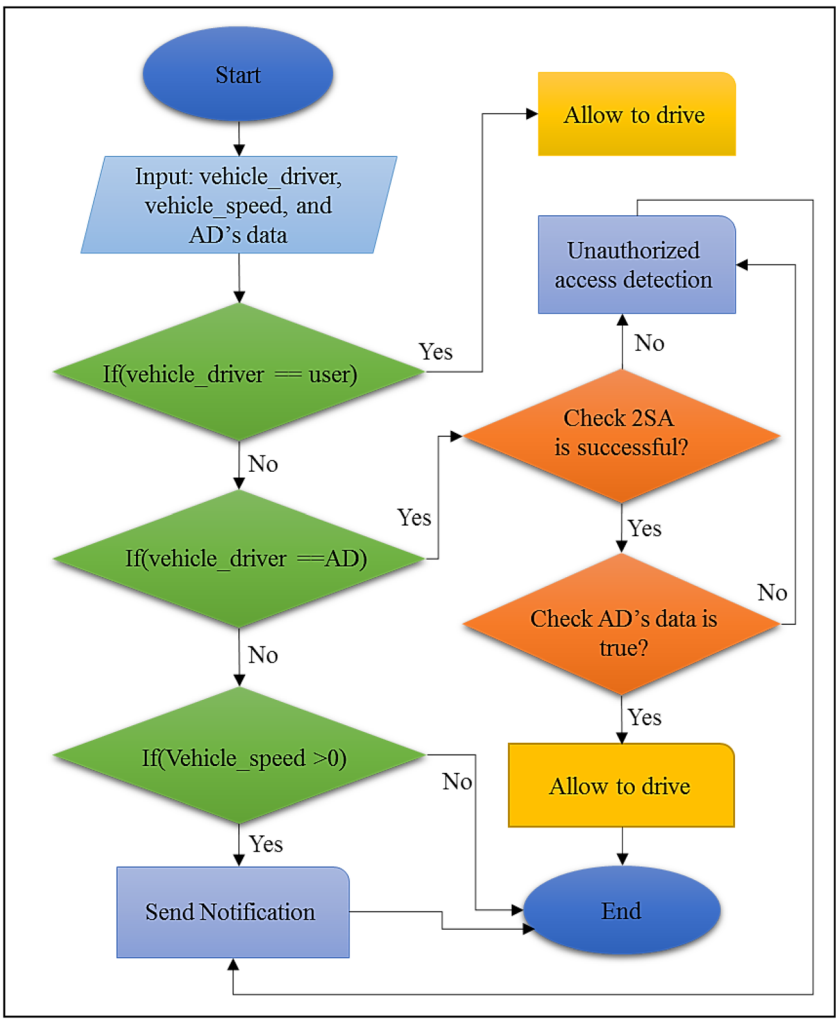


Fig. 8 Flowchart of the unauthorized access detection on a vehicle

3.2.2 Deactivation of the AD

Another feature of the proposed framework is the *Deactivation of the AD*. The user must need to deactivate the AD after completion of a driving session by the AD. So, the AD cannot drive the vehicle further. In this case, the user must need to complete the 2SA successfully. Then the user can deactivate the AD successfully using the self-destruct function [26]. This function of the smart contract will clear all contract's related data. Therefore, the user cannot access any features of the smart contract. So, the smart contract cannot get or save any data from the user account address. Thus, in the current session, the user cannot access any features of the proposed system. After completion of a driving session, the user can remove the AD's data and finally updates the vehicle status. Therefore, the application updates.

the vehicle storage with a null value. Thus, no one can able to access the vehicle as no authorization data is available to the vehicle. If the AD wants to drive the vehicle again, the user should deploy a smart contract, then add the AD's data and set the AD as a current driver finally. The process of the deactivation of the AD is shown in Fig. 7.

3.3 Unauthorized access detection

The process of unauthorized access detection is the 2nd part of the proposed framework. Every driver, including the user, has to authorize for driving the vehicle in each session. Only the authorized person can drive the user's vehicle. As per the user's request, data can be synchronized with the vehicle device to verify the selected AD. Therefore, the selected AD can drive the vehicle. The workflow diagram of unauthorized access detection has been shown in Fig. 8. The unauthorized access detection algorithm has also been presented in the following:

3.3.1 Unauthorized access detection algorithm

Algorithm 2: Unauthorized Access Detection

```
Input: vehicle_driver, vehicle_speed, AD_biometric, 2SA_success
1.  Start
2.  Restrict access to an account from any unauthorized user.
3.  if (vehicle_driver == user)
    then
        allow driving
    end
4.  if-else(vehicle_driver == AD) then
        if(2SA_success==true && AD_biometric == true)
        then
            allow driving
        end
        else
            send notification about the unauthorized access detection
        end
5.  else
        if (vehicle_speed > 0)
        then
            send notification about the unauthorized access detection
        end
        else
            nothing do
        end
    end
6.  End
```

3.3.2 Unauthorized access detection flowchart

4 Experiment results

This section focuses on the experimental results of the developed smart contract. We have experimented only with the two algorithms for the proposed framework.

4.1 Environment setup

Implementation of the smart contract has been experimented using the Remix-Ethereum IDE [27], which can provide an interface for deploying and testing smart contracts. The smart contract's code can be compiled and run on this interface. It is an open-source platform where we can write the smart contract's code in solidity language. Solidity is the most familiar language to write and test the smart contract's code. We have

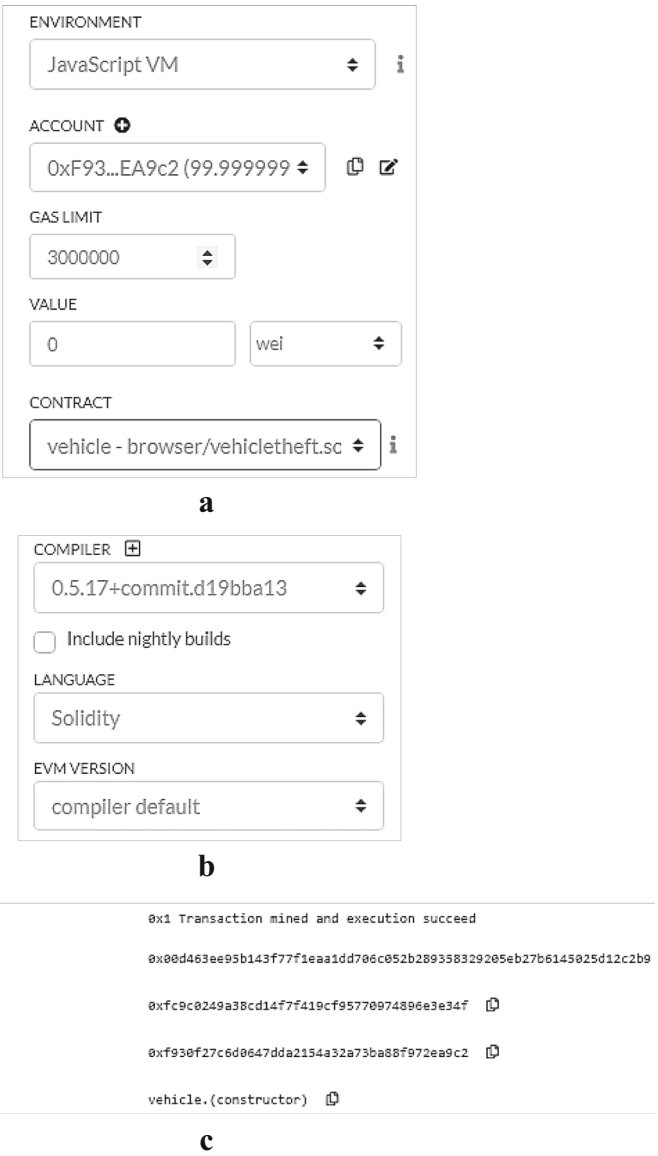


Fig. 9. **a** Environment setup. **b** Setup version of compiler and EVM. **c** Successful deployment of the smart contract

chosen the solidity compiler version ^5.0.17, which is shown in Fig. 9b. Figure 9a shows the environment setup of the smart contract named *vehicletheft.sol*. Figure 9c shows that the smart contract has been deployed successfully from the account address *0xF930F27C6D0647Dda2154a32A73bA88F972EA9c2* to *0xfc9c0249a38cd14f7f419cf95770974896e3e34f*.

4.2 2-step authentication

For the 2SA, the user has to login to the application site. If the login is successful, then the user will be completed the 1SA. We have taken the token *203,528* randomly. Figure 10a shows a token has been set by the user from account address *0xF930F27C6D0647Dda2154a32A73bA88F972EA9c2*. Figure 10b shows that the transaction of the token setting has been executed successfully. Figure 10c shows that the user has been authenticated using the correct token: *203528*. After providing the correct token, the user has been authenticated successfully, which is shown in Fig. 10d. Therefore, the user will be completed the 2SA. If the user provides the wrong token: *203526*, then the user cannot be authenticated that is shown in Fig. 10e.

4.3 Data insertion of an AD

The user needs to complete the 2SA for adding information of an AD. The AD can drive the vehicle after successfully inserting the data of the AD by the user. Figure 11a shows that the user inserts an AD's data using the correct token: *203528*. Therefore, the data has been inserted successfully, which is shown in Fig. 11b. Another side Fig. 11c shows that the user has been tried to insert data using a wrong token:

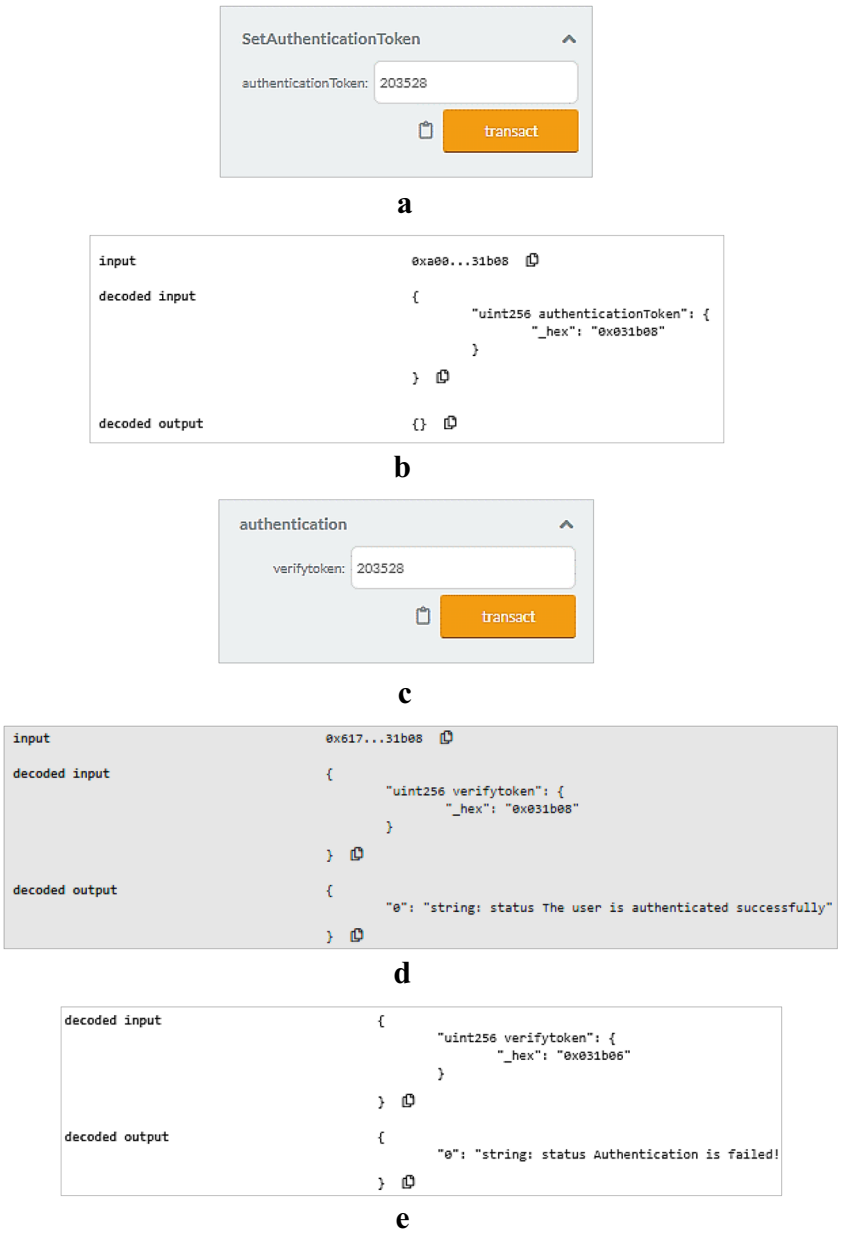


Fig. 10. **a** Token setting by the user. **b** Transaction of the token setting has been successfully executed. **c** User authentication using the correct token. **d** User has been authenticated successfully using the correct token. **e** User authentication has been failed for providing the wrong token

203529, but the transaction cannot be executed for providing the wrong token, which is shown in Fig. 11d.

4.4 Current driver setting

The AD can be set as a current driver by the user using the correct token. If the token is validated by the smart contract, then the AD will be set as the current driver. Therefore, the AD can drive the vehicle. Figure 12a and b are shown that the user has been set the current driver successfully using the correct token: *203528*. And, Fig. 12c and d are shown that the user cannot set the current driver using the wrong token: *203553*.

4.5 Deactivation of a driver

When a driving session has been completed, the user can clear the AD's data by using this feature. So, the AD cannot drive the vehicle in the future. Figure 13a shows that AD's data cannot be inserted after clearing the smart contract's data. Here, we have chosen a license number: 333 and tried to add data, but the data hasn't been added, and the AD cannot be set as a driver.

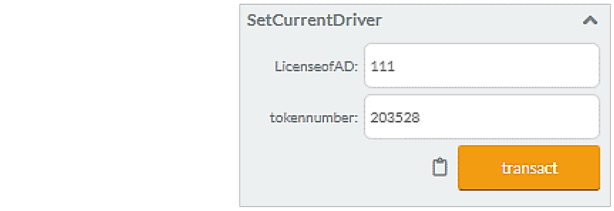
5 Performance analysis

In this paper, a vehicle theft detection framework has been proposed using blockchain and smart contracts. In this

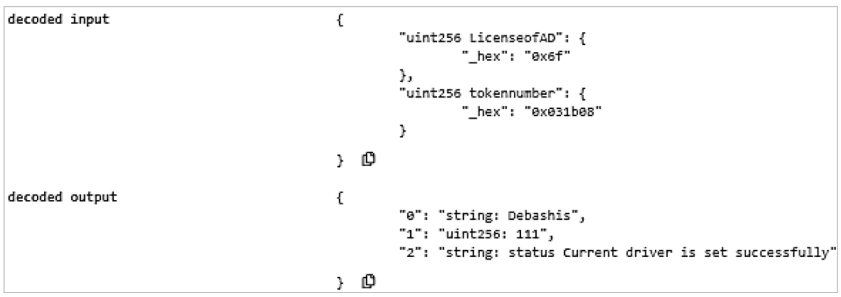
Fig. 11. **a** Data insertion using correct token. **b** Transaction of the data insertion has been executed successfully. **c** Data insertion using the wrong token. **d** Data cannot be inserted using the wrong token

framework, blockchain can provide a decentralized, transparent, and secure platform. The smart contract can authenticate a genuine user and detect unauthorized access. Combing these two, we have designed a vehicle theft detection framework to overcome vehicle theft issues. No one can alter the data stored in the blockchain. As smart contracts can authenticate a genuine user, the vehicle cannot be theft by an unauthorized person. For the proposed framework, we have introduced two algorithms: 2SA and unauthorized access detection. An analysis of these algorithms ensures that the proposed framework can improve the security of vehicles. The experimental results also show that the user can use the proposed framework securely. The 2SA ensures the secure accessibility of the application by providing the randomly token chosen by the user. The 2SA can give a few advantages such as user-friendly, availability, and inexpensive to set up and maintain the system. Any features of the proposed framework can be accessed after completing the 2SA in a session. Whenever the user wants to access these features, he has to complete the 2SA. Unauthorized users cannot access these features as they cannot be authenticated through the 2SA.

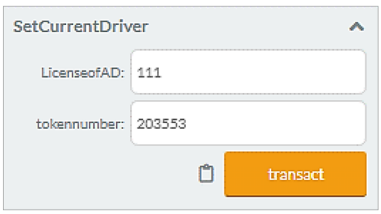
The block in the blockchain contains transactions that are shareable resources. The key feature of the blockchain is data security [10] that is mainly needed for the proposed framework. We have applied blockchain for a secure vehicle anti-theft system, where blockchain provides the security of data,



a



b



c

d

Fig. 12. **a** Current driver setting using the correct token. **b** Transaction of the current driver setting has been executed successfully. **c** Current driver setting using the wrong token. **d** Current driver setting has been canceled

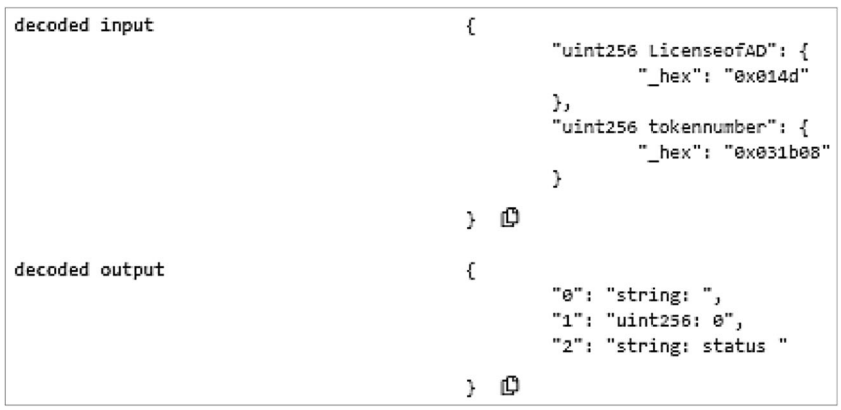


Fig. 13 **a** Driver cannot be set after deactivating the AD

Table 1 Comparison analysis of the BVATS with existing vehicle anti-theft methods

Methods Characteristic	GSM and GPS with Fingerprint Verification [2]	Using the Internet of things[1]	with a smartphone application[3]	BVATS
Cyber Attack	✓	✓	✓	✖
Personal Information Leakage	✓	✓	✓	✖
Data Security	✖	✖	✖	✓
Adequate Key Management	✖	✖	✖	✓
Transparency	✖	✖	✖	✓
Data Immutability	✖	✖	✖	✓
Data Availability	✖	✖	✖	✓
Privacy	✖	✖	✖	✓

✖ = Not Applicable, ✓ = Applicable

and the smart contract validates the authentication of a vehicle user. The smart contract can be deployed in the blockchain ledger to protect from unauthorized modification of the contract's code. It is possible to set privileges to access blockchain data. Since the smart contract can be used to access the stored data and store processing data, it is stored on the blockchain ledger. No one can modify the smart contract's code as the blockchain can provide an immutable ledger [28]. Thus, the smart contract is secured in this proposed framework, and the data stored in the blockchain ledger is nearly impossible to mangle [29, 30].

We found out the limitations of existing methods [1–3], analyzed existing issues [31, 32], and provided a comparative analysis with the proposed method. While consistent security and privacy in a traditional system can be impossible to obtain, blockchain may do so by the use of its intrinsic features to ensure security against fraudulent attempts. Since each connected node in the blockchain network has the same copy of the information, availability, privacy, and immutability of the data are presented in the proposed framework [33]. We compared the proposed method with existing methods that are shown in Table 1. A clear comparison of analysis has been prepared based on available features, characteristics, and properties of technologies used in existing methods.

Because the automobile industry is getting advanced rapidly, most of the advanced vehicles are offering keyless features. Moreover, if someone has a key used vehicle, it isn't necessary to confirm the driver using the biometric verification at the time of ignition. The doors of the vehicle can be opened by any means. Whether a biometric system is required to verify the driver, thus only keyless vehicles are considered in the proposed framework.

6 Conclusion and future work

In this paper, we have designed a vehicle theft detection framework for protecting vehicles. We have selected a decentralized platform like blockchain to overcome existing issues of a centralized system. The smart contract can enhance the security of an automated authentication process of the user with higher accuracy. Our proposal mainly concentrates on the security of the vehicle to guard it against unauthorized access and provides the privacy of available features of the application. The implementation of the proposed framework shows that it can provide a secure, decentralized, transparent, and trust environment for the security of vehicles as well as user privacy. In this paper, we have only tested on the designed smart contract based on proposed algorithms. In the future, we want to extend this research work by experimenting on data transfer from application to vehicle. We want to design a vehicle-to-vehicle (V2V) communication system for avoiding road accidents using blockchain, multimedia, and sensing device.

References

1. Ghasempour A (2019) Internet of things in smart grid: architecture, applications, services, key technologies, and challenges. Inventions 4:22. https://doi.org/10.3390/inventions4010022
2. Dey M, Arif MA, Mahmud MA (2017) Anti-theft protection of vehicle by GSM & GPS with fingerprint verification, 2017 International conference on electrical, computer and communication engineering (ECCE), Cox's Bazar 916-920. https://doi.org/10.1109/ECACE.2017.7913034
3. Liu B, Liu N, Chen G, Dai X, Liu M (2018) A low-cost vehicle anti-theft system using obsolete smartphone. Mob Inf Syst 2018: 16–16. https://doi.org/10.1155/2018/6569826
4. Kiruthiga N, Latha L, Thangasamy S (2015) Real time biometrics based vehicle security system with GPS and GSM technology. Procedia Comput Sci 47:471–479. https://doi.org/10.1016/j.procs.2015.03.231
5. Liu Z, He G (2005) Research on Vehicle Anti-theft and Alarm System Using Facing Recognition International Conference on Neural Networks and Brain. Beijing. 925–929. https://doi.org/10.1109/ICNNB.2005.1614771

6. Jaccard JJ, Nepal S (2014) A survey of emerging threats in cyber security. J Comput Syst Sci 80(5):973–993. https://doi.org/10.1016/j.jcss.2014.02.005
7. Gendrullis T, Novotny M, Rupp A (2008) A real-world attack breaking A5/1 within hours. Cryptographic hardware and embedded systems-CHES 2008. CHES 2008. Lect Notes Comput Sci 5154:266–282. https://doi.org/10.1007/978-3-540-85053-3_17
8. Ko E, Kim T, Kim H (2017) Management platform of threats information in IoT environment. J Ambient Intell Humaniz Comput 9(4):1167–1176. https://doi.org/10.1007/s12652-017-0581-6
9. Rui H, Huan L, Yang H, YunHao Z (2020) Research on secure transmission and storage of energy IoT information based on Blockchain. Peer-to-Peer Netw Appl 13:1225–1235. https://doi.org/10.1007/s12083-019-00856-7
10. Alotaibi B (2019) Utilizing Blockchain to Overcome Cyber Security Concerns in the Internet of Things: A Review. In: Utilizing Blockchain to overcome cyber security concerns in the internet of things: a review, in IEEE sensors journal, 19(23), 10953–10971. https://doi.org/10.1109/jsen.2019.2935035
11. Hong S (2020) P2P networking based internet of things (IoT) sensor node authentication by Blockchain. Peer-to-Peer Netw Appl 13:579–589. https://doi.org/10.1007/s12083-019-00739-x
12. Dimitrov DV (2019) Blockchain applications for healthcare data management. Healthcare Informatics Res 25(1):51–56. https://doi.org/10.4258/hir.2019.25.1.51
13. Chang Y, Iakovou E, Shi W (2019) Blockchain in global supply chains and cross border trade: a critical synthesis of the state-of-the-art, challenges and opportunities. Int J Prod Res:1–18. https://doi.org/10.1080/00207543.2019.1651946
14. Singh PK, Singh R, Nandi SK, Nandi S (2019) Managing Smart Home Appliances with Proof of Authority and Blockchain. Commun Comput Inf Sci:221–232. https://doi.org/10.1007/978-3-030-22482-0_16
15. Zhu X, Wang D (2019) Research on Blockchain Application for E-Commerce. Finance and Energy IOP Conf. Ser.: Earth Environ Sci. 252:042126
16. Zhang Y, Wen J (2017) The IoT electric business model: using blockchain technology for the internet of things. Peer-to-Peer Network Appl 10:983–994. https://doi.org/10.1007/s12083-016-0456-1
17. Li H, Pei L, Liao D, Sun G, Xu D (2019) Blockchain meets VANET: an architecture for identity and location privacy protection in VANET. Peer-to-Peer Netw. Appl 12:1178–1193. https://doi.org/10.1007/s12083-019-00786-4
18. Qian M, Gao H, Liu W (2018) Android Based Vehicle Anti-Theft Alarm and Tracking System in Hand-Held Communication Terminal. 2018 IEEE international conference on consumer electronics-Taiwan (ICCE-TW) Taichung. 1-2. https://doi.org/10.1109/ICCE-China.2018.8448426
19. Wei J, Chiu C, Huang F, Zhang J, Cai C (2019) A cost-effective decentralized vehicle remote positioning and tracking system using BeiDou navigation satellite system and Mobile network. J Wireless Com Network 2019:112. https://doi.org/10.1186/s13638-019-1436-y
20. Zheng H, Shao J, Wei G (2020) Attribute-based encryption with outsourced decryption in blockchain. Peer-to-Peer Netw Appl 13:1643–1655. https://doi.org/10.1007/s12083-020-00918-1
21. Wang S, Ouyang L, Yuan Y, Ni X, Han X, Wang FY (2019) Blockchain-enabled smart contracts: architecture, applications, and future trends. IEEE Trans Syst Man, Cybernetics: Syst 49(11):2266–2277. https://doi.org/10.1109/tsmc.2019.2895123
22. Dashora C, Sudhagar PE, Marietta J (2019) IoT based framework for the detection of vehicle accident. Cluster Comput 23:1235–1250. https://doi.org/10.1007/s10586-019-02989-z
23. Hlebiv O (2018) Ethereum Smart-Contract Storage https://applicature.com/blog/blockchain-technology/ethereum-smart-contract-storage. Accessed 30 Apr 2020
24. Solidity: Introduction to Smart Contracts (2017) https://solidity.readthedocs.io/en/v0.4.21/introduction-to-smart-contracts.html. Accessed 30 Mar 2020
25. Solomon MG (2020) Ethereum Smart Contracts: Tips for Handling Data in Solidity https://www.dummies.com/personal-finance/ethereum-smart-contracts-tips-for-handling-data-in-solidity/. Accessed 30 Mar 2020
26. Felker D (2018) Self Destructing Smart Contracts In Ethereum. https://articles.caster.io/blockchain/self-destructing-smart-contracts-in-ethereum/. Accessed 5 Jun 2020
27. Remix, Ethereum-IDE: Welcome to remix documentation (2019) http://remix.ethereum.org/. Accessed 15 May 2020
28. Liu J, Liu Z (2019) A survey on security verification of Blockchain smart contracts. IEEE Access 7:77894–77904. https://doi.org/10.1109/access.2019.2921624
29. Karbasi AH, Shahpasand S (2020) A post-quantum end-to-end encryption over smart contract-based blockchain for defeating man-in-the-middle and interception attacks. Peer-to-Peer Netw Appl 13:1423–1441. https://doi.org/10.1007/s12083-020-00901-w
30. Zheng B, Zhu L, Shen M et al (2018) Scalable and privacy-preserving data sharing based on Blockchain. J Comput Sci Technol 33:557–567. https://doi.org/10.1007/s11390-018-1840-5
31. Ko E, Kim T, Kim H (2017) Management platform of threats information in IoT environment. J Ambient Intell Humaniz Comput 9(4):1167–1176. https://doi.org/10.1007/s12652-017-0581-6
32. Jaccard JJ, Nepal S (2014) A survey of emerging threats in cybersecurity. J Comput Syst Sci 80(5):973–993. https://doi.org/10.1016/j.jcss.2014.02.005
33. Paik HY, Xu X, Bandara HMND, Lee SU, Lo SK (2019) Analysis of data Management in Blockchain-Based Systems: from architecture to governance. IEEE Access 7:186091–186107. https://doi.org/10.1109/access.2019.2961404

Debashis Das is currently working as a university research scholar at Kalyani University. He has completed the Master of Technology in Computer Science and Engineering in 2018 from Kalyani Government Engineering College. He has completed a Bachelor of Technology in Computer Science and Engineering in 2015 from the Government College of Engineering and Leather Technology. His research interests are including Cloud Computing, Internet of Things (IoT), Distributed Computing, and Blockchain Technology. Now, He is working on Blockchain technology.

Sourav Banerjee holds a PhD degree from the University of Kalyani in 2018. He is currently an Assistant Professor at Department of Computer Science and Engineering of Kalyani Government Engineering College at Kalyani, West Bengal, India. He has authored numerous reputed journal articles, book chapters and International conferences. His research interests include Big Data, Cloud Computing, Cloud Robotics, Distributed Computing and Mobile Communications, IoT. He is a member of IEEE, ACM, IAE and MIR Labs as well. He is a SIG member of MIR Lab, USA. He is an Editorial board member of Wireless Communication Technology. He is the reviewer of IEEE Transactions on Cloud Computing, Wireless Personal Communications, Journal of Ambient Intelligence and Humanized Computing, Journal of Computer Science, Journal of Supercomputing, Future Internet, International Journal of Computers and Applications, Personal and Ubiquitous Computing, Innovations in Systems and Software Engineering, etc. He is connected with various international events, like, Workshop on Security and Privacy in Distributed Ledger Technology (IEEE SP-DLT). He is a lead guest editor of Complex and Intelligent Systems Journal (Springer, SCI, IF: 3.79) for the Special Issue on "Advancement and Trends in Green Cloud Computing, Blockchain and IoT for Modern Applications and Systems". A number of worldwide research scholars are attached with him.

Utpal Biswas received his B.E, M.E and PhD degrees in Computer Science and Engineering from Jadavpur University, India in 1993, 2001 and 2008 respectively. He served as a faculty member in NIT, Durgapur, India in the department of Computer Science and Engineering from 1994 to 2001. Currently, he is working as a professor in the department of Computer Science and Engineering, University of Kalyani, West Bengal, India. He has over 130 research articles in different journals, book chapters and conferences. His research interests include optical communication, ad-hoc and mobile communication, semantic web services, E- governance, Cloud Computing etc.