Auto Block IoT: A Forensics Framework for Connected Vehicles

P A Abhay, N V Jishnu, K T Meenakshi, P S Yaswanth and Abin Oommen Philip

Department of Computer Science Engineering Toc H Institute of Science and Technology, India, Kerala, Ernakulam, Arakunnam 682313

abhaypa1925@gmail.com

Abstract. The increase in the number of vehicles has led to the increase in number of accidents. The rise in connected vehicles and autonomous vehicles will increase the level of forensics and investigations required in case of an accident. A hindrance to the investigation is the reliability of data. This paper proposes a blockchain framework to ensure that the vehicular data at the time of accident is stored in a tamper-proof and reliable way for further analysis by various stakeholders. A test bed using IoT, IPFS and Ethereum test network is proposed and evaluated to study the feasibility of the proposed approach.

1. Introduction

As many as 17 people died in 55 road accidents per hour on average last year, with half of them coming under the age group of 18-35. While overall road accidents declined by 4.1 percent, the facilities increased by 3.2 percent, i.e., more than 400 people lost their lives daily on the road. A total of 4,80,652 road accidents took place in India in the year 2016 resulting in loss of 1,50,785 lives and inflicting serious injuries on 4,94,624 persons [1]. Effective investigation and settlement of accident cases is necessary for insurance and law purposes. At present the data related to accidents are investigated and handled manually which decreases trust as the process is corruptible.

As vehicles become connected and autonomous with the advent of the Internet of Vehicles [2] the need for accident forensics would increase as drivers may not be solely responsible for the accidents. Accidents may occur due to software and hardware errors too. Several changes have been brought out in Intelligent Transportation Systems to solve the mysteries of vehicle crashes for decades [3-9]. Event data recorders are present in few vehicles and track vehicular data such as speed, acceleration, braking, steering and air-bag deployment before, during and after a crash. Despite their presence in cars since the mid-1990s, they haven't fulfilled their potential because the EDR data is not tamper-proof, limited and is erasable.

In an ever-evolving era of connected and autonomous vehicles, rather than merely storing the vehicular data into data recorders, there is an urgent need to tamper proof these data. Most cars being manufactured today has, EDRs or Event Data Recorders which log vehicular data seconds before and seconds after a possible event. Connected and autonomous vehicles will contain software and hardware from multiple vendors. The cause of the accident may be a fault in a software or hardware component and the vendor may try to manipulate the EDR data to avoid claims [7].

In the future, vehicles and road infrastructures would be incorporated with OBUs (on-board units), RSUs (Road-side units), DSRC (Dedicated Short-Range Communications) etc which has the capability to implement vehicle-to-vehicle communication and vehicle - to - infrastructure (such as traffic lights,
railroad crossing etc) communication [2] and thus avoid accidents. The information gathered through such communication can be applied in post-accident scenarios to provide valuable evidence regarding the incident, which can be utilized by manufacturers, law enforcement authorities etc. The data obtained from OBD and RSU is not tamper proof by itself, they can be manipulated by the vendors for hiding their faults or to divert the investigation if stored in cloud or database. By combining a blockchain and designing a fragmented ledger to store all the vehicular data, a much more comprehensive and re-creatable data can be forwarded to the forensic investigators from which they can find stronger, trustworthy, tamper-proof and reliable evidence about the incident.

Our proposed system is a blockchain based forensics framework for the storage and retrieval of vehicular data related to an accident. Thus, enabling tamper proof storage of accident-related data. We propose a framework for storage of accident-related data and later retrieval of the data for forensics by different stakeholders. We implemented an IoT testbed incorporating sensors like triple axis Accelerometer, triple axis Gyroscope and GPS breakout to simulate vehicular incident data. This data is stored in IPFS [10] via INFURA and the hash from IPFS is stored to Blockchain via Ethereum framework. Ganache is used for setting up an Ethereum Blockchain workspace or local network. The system is evaluated in terms of feasibility of the approach. In future the vehicles can be incorporated with a secure IoT hardware device capable of being triggered by an accident likely scenario and transmit preset vehicle parameters onto Blockchain.

The paper is structured as follows: in section II background study related to the technology and terms used for understanding the framework is presented. Section III details the design of the proposed system. Section IV covers the implementation of the system. Section V gives details about the evaluation and feasibility analysis of the system. Section VI concludes and presents the future scope.

2. Background Study
In this section we briefly discuss the

- Traditional Vehicular Accident Forensics approach.
- Forensics based on Event Data Recorders and On-Board units in vehicles.
- Study on Technologies used in the proposed framework and related work.

2.1. Traditional Vehicular Forensics approach
Traditionally accident analysis is performed in four steps:

- Fact gathering: After an accident, a forensic process is initiated to gather all possibly relevant facts that may contribute to understanding the accident and can be used as evidence. The witness statements, photographs of scenes are also recorded.
- Fact Analysis: Once the forensic process is completed all the relevant facts are gathered and the history of the accident is reconstructed. Then its consistency and plausibility are checked.
- Conclusion Drawing: Conclusions can be drawn about causation and contributing factors given that the accident history is informative to a sufficient extent.
- Countermeasures: In some cases, the development of countermeasures are made to prevent further accidents of the same kind.

The problems associated with manual traditional approach is the trustworthiness of the evidence presented or lack of evidence. As vehicles become connected the fault can also be associated with the hardware and software components of the vehicle. This paved way to digital forensic approaches involving Event Data Recorders and On-Board Devices.

2.2. Event Data Recording and On-Board Diagnostic
Event Data Recorders (EDR) and On-Board Diagnosis (OBD): EDR’s informally named as “black box” is a device placed in vehicles in order to collect data related to crashes and incidents. In case of a dispute, investigators come up with the most probable setup. The digital data recorded by the EDR is widely used as supporting evidence in the investigation for reconstructing the accident scene. When a triggering event occurs, some among those events are the airbag deployment, sudden speed changes above the
threshold etc EDR captures and stores the state of the vehicle. It is known that EDR data is extracted by the investigators through the onboard diagnosis (OBD) port in case of an accident [5]. The data recorded varies by model, but commonly recorded information may include:

- Pre-crash vehicle dynamics and system status
- Driver inputs
- Vehicle crash signature
- Restraint usage/deployment status
- Post-crash data, such as the activation of an automatic collision notification (ACN) system.

Currently, EDRs are widely installed and used by vehicle manufacturers, insurance companies, law enforcement agencies, and researchers. Insurance companies adopt EDRs as a way to gain better insight into driver behaviour analysis. Individuals who buy insurance may install the monitoring system which may decrease the negative impact of recording information. However, they have to trade it off against the loss of privacy by which insurance companies calculate insurance premiums according to the information derived from the EDR.

Many accidents reconstructionist prefer to rely on EDR data to provide an unbiased measure of pre-impact actions and impact speed. However, there are times that EDR information should not be relied upon, or require in-depth interpretation, such as when:

- The EDR data might contradict physical evidence.
- The data might be out of the common driving range (low or high).
- There might be a discontinuity in the data (flat lines, spikes, etc).
- The ownership of the data is disputed, or the chain of evidence is suspect.

Moreover, the EDR device being an EEPROM is erasable. Thus, the data obtained from EDR devices cannot be considered as an immutable source of evidence.

2.3. Study on Technologies used in Proposed System and Related work

Blockchain as an Immutable Ledger. Blockchain[11] was traditionally used for financial applications. Of late its being utilized as a distributed ledger to store data and transactions in an immutable manner. The hashing of blocks as a chain provides the immutability property. The blockchain can be private/public or permissioned/permissionless based on who is allowed to read/write the data. We propose using blockchain for tamper proof storage of accident-related data. Various stakeholders like transportation department, police, law, insurance, auto manufacturers can access the post-accident data based on privileges provided. If data is stored in public blockchain it is accessible to all, further encryption must be provided to limit the access. If data is stored on a private blockchain we can limit permissions to only authorized users and assign different access privileges. Authors [12-14] have proposed various methods and frameworks related to storing vehicular accident data onto blockchain. Inspired from all the frameworks we propose a testbed to evaluate the feasibility of deploying such a system via smart contracts on Ethereum blockchain.

2.3.1. Ethereum. It is a decentralised blockchain-based computing platform which is open-source. The powerful feature which enables it to perform more functions than cryptocurrencies is the smart contract(scripting) functionality. Ether is the cryptocurrency in Ethereum platform, ether is also given as a reward to mining nodes for computations performed and is the only currency accepted in the payment of transaction fees on the platform [15]. We have utilised Ethereum as a platform to evaluate the framework capabilities.

2.3.2. IPFS. The Interplanetary File System (IPFS) is a distributed peer-to-peer protocol for storing and sharing data in a distributed file system. In order to uniquely identify files content-addressing is used in IPFS. IPFS is a decentralised system when a file is uploaded or added to the system it is broken down into smaller pieces and then distributed in the network. So, users in the system will each have a small portion of the original data, creating a resilient system of file storage and sharing. The files are accessed by users through the content address and other peers in the network can find and request that content
from any node who has it using a distributed hash table (DHT)[10]. When an accident occurs an enormous amount of data in the form of sensor data, image, video can be generated related to the accident. Storing all this data onto blockchain is not feasible in terms of cost and scalability. Thus, the data can be stored onto the IPFS and the hash retrieved is put on the blockchain. The data can be retrieved from the IPFS using hash addressing.

3. Proposed Framework Design

Figure 1. Proposed framework design showing the data flow and transaction flow.

Figure 1 shows the overview of the proposed system. Here every vehicle is equipped with an OBD unit consisting of various sensors like accelerometer, gyroscope, GPS etc. These sensors constantly monitor the readings. A buffer collects these data and refreshes every 10 seconds. The refresh time can be set as per requirement. When an accident occurs or an accident event is triggered, the sensor data is sent to Infura where the data will be stored to public IPFS node [13]. The data will be broken into smaller pieces and are stored distributedly in IPFS nodes. IPFS returns a hash for the data stored which is added to the Ethereum network. The data distributed among the nodes in the network can be accessed by the use of DHT (Distributed Hash Table). There are different stakeholders (Forensics department, Insurance agencies, manufacturers etc) who will be accessing these data. The stakeholder can interact with the Ethereum network through Metamask browser extension. Metamask helps the stakeholders to communicate with the Ethereum network. An application interface has been developed for stakeholders to view and analyze the forensic data.

There are mainly two kinds of flow in this system as depicted in figure 2.

- The control flow for Block Writing: When an accident is triggered the data is sent to IPFS for storage. The hash of the stored data is sent to the ethereum network via Smart Contract. The data gets added to the Ethereum network as a transaction in block. The block gets added after the consensus process. This hash is stored inside the Ethereum network making the block size minimum as compared to storing the whole accident data in blockchain. The data can be
encrypted and put onto the blockchain, so that only authorized stakeholders access the data. Access based controls can be set on public blockchain as suggested by authors in [14]

- The Control Flow for Data Retrieval: There are three main stakeholders. They request content from the ethereum network with the help of Metamask browser extension. The ethereum network validates the hash. The EVM (Ethereum virtual machine) communicates with the IPFS node to retrieve the data. The EVM reads the hash and requests the file from IPFS. The file is provided to the stakeholders. The file is encrypted and only those authorised can view the content. Indexing mechanism is used to retrieve the required file by specifying search parameters like vehicle id, road segment id, date etc.

Airbag activation, sudden application of brakes etc can act as a trigger in accident detection. Machine learning models can be developed for efficient detection of incidents.

**Figure 2.** Diagrammatical representation of interaction between write and read modules of blockchain.

4. Implementation details

As an initial testbed, mobile phone devices are used to take readings from sensors like accelerometer, gyroscope and GPS. Most modern mobile phones are enabled with all these sensors. This is an ongoing project, and the initial framework and implementation details are presented. The aim is to develop a secure event detection device that can be incorporated into vehicles that detects and triggers accidents and pushes the incident data onto the blockchain.

The accelerometer and the gyroscope sensors will return three-axis readings (x, y, z coordinates). GPS will return the coordinates of the accident location. A buffer is maintained for collecting the data. This buffer refreshes every 10 seconds. For the initial testbed, here the readings from the gyroscope's z-axis are used as triggers. Threshold value is set to Threshold < 9.8 or Threshold > 10.1 for z axis of gyroscope. This is used to trigger a sudden acceleration or deceleration event. Other event triggers are being incorporated as the work progresses.

When the reading crosses the threshold, the trigger happens. So, when the threshold is met, data in the buffer is written to the IPFS network, a second buffer with data for the next 5 seconds is also added to the network. The UI showing mobile phone sensor and GPS readings is shown in figure 3.

Accident detection happens when the values in the accelerometer, gyroscope etc exceed a certain threshold value. Here the threshold value of the z-axis of the gyroscope is given more importance as the sudden change in z value indicates vehicles have fallen or accidents have occurred. Once the threshold is reached, the app makes API calls to address the data collected from the vehicle through IPFS. The data is stored to a public IPFS network via INFURA. As discussed, IPFS helps in creating the encrypted hash file. When a data file is hashed, the result is a fixed-length string that is unique and particular to that file. This hash is stored in the blockchain. Each time a transaction is done a predefined number of
ethers is decremented. The ethereum test network is provided by Ganache[16] in the Truffle suite, which provides a personal blockchain for the development of rapid Ethereum.

![Figure 3. UI of the mobile application used to collect the test data, such as GPS, accelerometer and gyroscope readings.](image)

The retrieval part where stakeholders access the data is implemented by coding to the smart contract. The stakeholders can login to their interface and request data. Their request to the blockchain network is done with the help of Metamask extension. Metamask communicates with the ethereum network and returns the hash. Once the hash is known, the data can be retrieved. The data will be available in the hash location of IPFS. But this does not mean anyone with the hash value can access the data. Since the hash was added after encryption only those entitled can access the data. By encrypting the file itself with the public key of the stakeholder, the access to certain stakeholders is granted. This is coded into the smart contract via solidity. The UI of the login page for stakeholders is shown in figure 4.

This implementation gives only an initial feasibility result. The work is part of an ongoing project. A secure device that can be incorporated in vehicles along with OBD port capable of detecting incidents
using machine learning and analytics and transmitting relevant incident data onto blockchain is being developed.

![Image](image-url)

**Figure 4.** UI of the Login Page for stakeholders to login.

5. Analysis
The initial testbed was developed in Remix Ethereum IDE which is an IDE that supports testing, debugging and deploying of smart contracts. Here, the smart contracts are implemented using solidity language. The testing environment is set-up in Javascript Virtual Machine which is defined before the compilation and it provides the account for deployment and testing of smart contracts using the local ethereum network.

On achieving an error-free compilation, the smart contract will be deployed. Here in the smart contract, an array structure is used to store the vehicle number and the IPFS hash. In solidarity, arrays can have a dynamic size, therefore a variable is used to keep a count on the blocks being written. A sample of a transaction retrieved is shown in figure 5.

For analysis purposes we evaluate the test bed in terms of time to store and access data, deploy smart contract, cost of transmitting and storing IPFS hash etc. The analysis done, parameters used and values obtained are presented in table 1.

A trigger is generated by the application when the z axis of the gyroscope detects a value beyond the threshold. The values in terms of time and cost justify the feasibility of the framework.
Table 1. Analysis of the various functions in terms of time and cost.

| Analysis                                                                 | Parameter                                      | Value                       |
|--------------------------------------------------------------------------|------------------------------------------------|-----------------------------|
| Trigger based on z axis of gyroscope                                     | Threshold outside the specified range is considered as trigger | Gyroscope z axis value $< 9.8 || > 10.1 |
| Deployment of Smart Contract to Test network                             | Gas Cost                                       | .0089452 ETH                |
| Deployment Time for Smart Contract                                       | Time in Sec                                    | 2 Sec                       |
| Time taken to upload and retrieve IPFS hash                               | Time in secs                                   | 1 Sec                       |
| Time taken to write the incident hash retrieved from IPFS to blockchain via Smart contract | Time in Milliseconds                           | 125 ms                      |
| Cost of writing an incident data onto blockchain                          | Gas Cost                                       | 0.010629 ETH                |
| Time to retrieve data from blockchain by stakeholders                    | Time in Sec                                    | 3 Sec                       |
Figure 5. Transaction details of a sample block retrieved.

6. Conclusion and future work
A framework is proposed for storing vehicular accident-related data onto blockchain followed by retrieval by various stakeholders for forensic analysis, case settlement and insurance purposes. An initial test bed is proposed to evaluate the feasibility of the framework. The initial results are presented justifying the feasibility. The results form an initial part of the project of developing a secure hardware device integrated in vehicle along with OBD capable of detecting incidents in real time and transmitting accident evidence onto blockchain for later analysis. Further plans to expand the project involve:

- Developing and evaluating the framework in terms of scalability and efficiency on different blockchain platforms like Hyperledger, IOTA Tangle
- Developing a secure hardware device that can be incorporated into the vehicle during manufacturing capable of detecting incidents using Machine learning algorithms.
- Using the stored data to reconstruct the accident in 3D and aid visualisation.

References
[1] NDTV, 17 people died in road accidents per hour in India last year government, Press Trust Of India, September 06, 2017, 05:24 pm IST, Accessed on Dec 1st, 2020. [Online]. Available: https://www.ndtv.com/india-news/17-people-died-in-road-accidents-per-hour-in-india-last-year-government-1746849
[2] Abin Oommen Philip and Dr. Saravanaguru RA.K, A Vision of Connected and Intelligent Transportation Systems. International Journal of Civil Engineering and Technology, 9(2), 2018, pp. 873-882.
[3] Y. Yuan and F. Wang, Towards blockchain-based intelligent transportation systems, 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), Rio de Janeiro,
2016, pp. 2663-2668, doi: 10.1109/ITSC.2016.7795984.

[4] T. Jiang, H. Fang and H. Wang, Blockchain-Based Internet of Vehicles: Distributed Network Architecture and Performance Analysis, in IEEE Internet of Things Journal, vol. 6, no. 3, pp. 4640-4649, June 2019, doi: 10.1109/JIOT.2018.2874398.

[5] M. S. U. Alam, S. Iqbal, M. Zulkernine and C. Liem, Securing Vehicle ECU Communications and Stored Data, ICC 2019 - 2019 IEEE International Conference on Communications (ICC), Shanghai, China, 2019, pp. 1-6, doi: 10.1109/ICC.2019.8762043.

[6] Rakesh Shrestha, Rojeena Bajracharya, Anish P. Shrestha, Seung Yeob Nam, A new type of blockchain for secure message exchange in VANET, Digital Communications and Networks, Volume 6, Issue 2, 2020, Pages 177-186, ISSN 2352-8648, https://doi.org/10.1016/j.dcan.2019.04.003.

[7] Zubair A. Baig, Patryk Szewczyk, Craig Valli, Priya Rabadia, Peter Hannay, Maxim Chernyshhev, Mike Johnstone, Paresh Kerai, Ahmed Ibrahim, Krishnun Sansurooah, Naeem Syed, Matthew Peacock, Future challenges for smart cities: Cyber-security and digital forensics, Digital Investigation, Volume 22, 2017, Pages 3-13, ISSN 1742-2876, https://doi.org/10.1016/j.diin.2017.06.015.

[8] H. Mansor, K. Markantonakis, R. N. Akram, K. Mayes and I. Gurulian, Log Your Car: The Non-invasive Vehicle Forensics, 2016 IEEE Trustcom/BigDataSE/ISPA, Tianjin, 2016, pp. 974-982, doi: 10.1109/TrustCom.2016.0164.

[9] Dennis K. Nilsson and Ulf E. Larson. 2008. Conducting forensic investigations of cyber attacks on automobile in-vehicle networks. In Proceedings of the 1st international conference on Forensic applications and techniques in telecommunications, information, multimedia and workshop (e-Forensics '08). ICST (Institute for Computer Sciences,Social-Informatics and Telecommunications Engineering), Brussels, BEL, Article 8, 1–6.

[10] IPFS + Blockchain = Decentralised File Storage https://medium.com/coinmonks/ipfs-blockchain-decentralised-file-storage-9ef3a1fa307b

[11] Satoshi Nakamoto, Bitcoin: A Peer-to-Peer Electronic Cash System, https://bitcoin.org/bitcoin.pdf

[12] M. Cebe, E. Erdin, K. Akkaya, H. Aksu and S. Uluagac, Block4Forensic: An Integrated Lightweight Blockchain Framework for Forensics Applications of Connected Vehicles, in IEEE Communications Magazine, vol. 56, no. 10, pp. 50-57, OCTOBER 2018, doi: 10.1109/MCOM.2018.180013

[13] H. Guo, E. Meamari and C. Shen, Blockchain-inspired Event Recording System for Autonomous Vehicles, 2018 1st IEEE International Conference on Hot Information-Centric Networking (HotICN), Shenzhen, 2018, pp. 218-222, doi: 10.1109/HOTICN.2018.8606016.

[14] Philip, A. Oommen and Saravanaguru, RA K. Secure Incident & Evidence Management Framework (SIEMF) for Internet of Vehicles using Deep Learning and Blockchain Open Computer Science, vol. 10, no. 1, 2020, pp. 408-421. https://doi.org/10.1515/comp-2019-0022

[15] Vitalik Buterin, Ethereum whitepaper, https://ethereum.org/en/whitepaper/ (accessed Dec 9,2020)

[16] Documentation about Ganache “https://www.trufflesuite.com/docs/ganache/overview”