MACHINE LEARNING AND BLOCKCHAIN UTILIZED AUTHENTICATION FOR VANET

A Thesis

by

Hüseyin Tuncel

Submitted to the
Graduate School of Sciences and Engineering
in Partial Fulfillment of the Requirements for
the Degree of

Master of Science

in the
Department of Computer Science

Özyeğin University
June 2024

MACHINE LEARNING AND BLOCKCHAIN UTILIZED AUTHENTICATION FOR VANET

Approved by:

Asst. Prof. Kubra Kalkan Cakmakci,
Advisor
Dept. of Computer Science
Özyeğin University

Prof. Dr. Ali Cengiz Begen
Dept. of Computer Science
Özyeğin University

Assoc. Prof. Dr. Muge Sayıt
Dept. of Information Technologies
Ege University

Date Approved: May 24, 2024

To Zeynep, Topak, Tırmık and Köpük.

ABSTRACT

Vehicular Ad-Hoc Networks (VANETs) are specialized mobile networks that facilitate communication between vehicles and servers during driving. Through this communication, drivers connect to servers using various applications and share personal information with these applications. In VANET systems, authentication methods are used for security to access applications containing sensitive information on vehicles. The most common method for authentication in VANETs involves analyzing the driving behavior of the driver using sensor data collected from the vehicle to verify the driver's identity. This technique have some limitations which could stem from vehicle hardware, database capacity, or information security issues. Examples of limitations include the performance degradation of algorithms due to the usage of raw data, processing and capacity constraints of cars with low hardware, database capacity limitations, and the processing burden imposed by blockchain technology. Existing works does not consider authentication for a driver who uses a different brand car. This is a very common case for even a spouse who have different brand cars. In order to provide this property, various brands should provide data. This condition comes with a problem that different manufacturing companies should share data and agree on the conditions. This trusted environment can be provided by blockchain.

In our study, we proposed a framework on VANET where different car manufacturers coexist in the same network in a trusted environment. Additionally, we enable drivers to securely access past driving data when they desire to use a car from a different brand. Blockchain has been integrated to establish a trusted environment for information exchange among different vehicle manufacturers. The integration of blockchain has created a constraint in recording raw driving data, which requires a

large capacity. Therefore, in our study we applied Correlation-based Feature Selection (CFS) to the raw driving data collected at the end of the drive, resulting in the creation of a reduced dataset. Recording the reduced dataset on the blockchain for use as historical data in identity verification has decreased the capacity requirements on the blockchain and increased the Transaction per Second (TPS) value of the blockchain. The impact of CFS on identity verification algorithms was investigated, and it was observed that the performance of the algorithms improved. With this dataset, a %99 accuracy was achieved using the kNN algorithm. We also provided security protocols which provides secure communication between the entities in the framework. In our framework, blockchain is provided in between manufacturing peers of the brands. The RSU (Road Side Units) of VANETs are communicating with the manufacturing servers and these servers are communicating with the peers of the blockchain system. For this leveled structure, we provided security protocols in between different entities. Security proofs of these protocols are provided through AVISPA tool.

ÖZETÇE

VANET ağaları, hareketli nesneler olan araçlar ile sunucuların haberleşmesi için tasarlanan özel bir ağ yapısıdır. Bu haberleşme imkanını sağlayan ağ yapısı sayesinde sürücüler araçlarındaki uygulamalar ile sunuculara bağlanabilmekte ve kişisel bilgilerini kullanarak kendi hesaplarına erişebilmektedirler. Bu erişim esnasında sürücünün kişisel hesaplarını korumak için kimlik doğrulama yöntemleri kullanılır. Araçlarda uygulanılan kimlik doğrulama yöntemlerinden en yaygını sürücünün sürüş davranışını analiz edilerek kimliğinin doğrulanmasıdır. Bu yöntemi araçlara ve VANET sistemlerine entegre edebilmek için sistemin getirdiği kısıtlar bulunmaktadır. Bu kısıtlar aracın düşük donanım gücüne sahip olmasından, veri tabanının yüksek kapasiteler ihtiyaç duymasından ve bilgi güvenliğininin korunması gerekliliğinden kaynaklı olabilmektedir. Bu kısıtlara örnek olarak araç gibi düşük donanıma sahip cihazların yüksek boyutlu ham sürüş verilerini işleyememesi, blok zincirine dahil edildiği takdirde araca gelen yüksek işlem yükleri kaldıramaması örnek verilebilir. Bu kısıtlarla ilgili literatürde yapılan çalışmalar incelendiğinde sürücünün farklı marka araçlar kullandığında kimlik doğrulamasının nasıl yapılacağı incelenmemiştir. Gerçek hayatta çok rahatlıkla karşılaşabileceğimiz bir sürücünün farklı marka araçları kullanma ihtimali için farklı markaların da birbirleri ile bilgi paylaşımı yapması gerekmektedir. Farklı markaların birbirleri ile bilgi paylaşması ve bu paylaşılan bilgide anlaşmaları için güven ortamı sağlanmış bir yapı gereklidir. Bu yapı blok zinciri entegrasyonu ile sağlanabilir.

Çalışmamızda, bir sürücünün farklı marka araçlar kullanırken tüm markalardaki geçmiş sürüş verilerine erişebilmesi için aynı ağda bulunacak araç üreticileri için güven ortamı sağlanmış bir VANET yapısı sunulmuştur. Bu güven ortamını sağlamak için blok zinciri entegrasyonu yapılmıştır. Blok zincirinde yüksek veri boyutlarına sahip

ham sürüş verisinin kayıt edilmemesi için çalışmamızda ham sürüş verisine Correlation Based Feature Selection-(CFS) uygulanmıştır. CFS ile elde edilen çok daha düşük boyuttaki yeni verinin blok zincirinde tutulması hem blok zincirinin kapasite ihtiyacından doğacak kısıtı engellemiş hem de CFS verisi ile eğitilen makine öğrenim algoritmalarının daha keskin sonuçlar vererek kimlik doğrulamanın doğruluğunu arttırdığı gözlemlenmiştir. Test aşamasında CFS ile eğittiğimiz kNN algoritması %99 doğruluğa ulaşmıştır. Sonuç olarak bu çalışma ile sürücülerin farklı marka araçlardaki geçmiş sürüş verilerine ulaşabileceği ve farklı marka araç üreticilerinin birbirleriyle güvenle veri paylaşabileceği bir VANET yapısı sunulmuştur. Ayrıca sunduğumuz yapıda ağ bileşenlerinin birbirleri ile güvenli haberleşmesi için aralarındaki haberleşme protokolleri sunulmuş ve protokollerin güvenli olduğu AVISPA TOOL yardımı ile test edilmiştir.

ACKNOWLEDGEMENTS

To begin with, I would like to express my absolute gratitude to my thesis supervisor, Prof Dr. Kübra Kalkan, for her invaluable tutelage, support, and encouragement throughout the research and writing process. It was great fun and quite exciting working with her. During the research process, her meticulous attention to detail and precision provided me with a vision for future work, particularly in terms of research methodology. Despite all the challenges we faced during the pandemic, her motivation and support played a significant role in completing our research.

Moreover, I also want to extend my gratitude to my dear spouse, Zeynep. The motivation and support she provided during my master's degree and research process were priceless. I also congratulate her on completing her master's degree and her new job in her career this year. I will be forever grateful for all the support she has given me, and I will gladly extend it to her in her future endeavors.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

1.1 Motivation

The internet of things and the advancing technology of digital transformation make this structure a global information network to which all devices can connect to each other [1], [2]. The tendency towards digital transformation is increasing in institutions that serve physical devices with the internet of things technology; These institutions are moving from a production-based ecosystem to a software services-based income ecosystem. For this reason, IoT technology is a pillar of the acceleration of digitalization.

Many product sectors such as automobiles have also been affected by this technological change. Approximately 2 billion cars connected to the Internet will be active in traffic in the next 20 years [3]. Vehicles with this technology will be connected to service providers and manufacturers over the internet and will provide many benefits to drivers. In addition to vehicle manufacturer companies, banks, fuel stations, road services, and toll services will also be active in this vehicle network connected to the internet. The network created by vehicles connected to the Internet is called VANET (Vehicular Ad Hoc Network). VANETs are network types where two or more vehicles communicate with each other or with RSU (Road Side Units). The RSU is a device that is settled on the nearside of roads. It provides a specialized wireless network to serve as a connection point for mobile devices, such as cars, to communicate with the server.

As in every network system, VANET systems are also attractive for attackers. Among these attack types, Sybil attacks, Node impersonation, Message suppression,

Alteration, Relay, and GPS spoofing attacks can be counted as moderate to high attacks. Against these attacks, precautions can be taken with the authentication method, which is one of the most important security services, that ensures the message is sent by the real client [4]. A Sybil attack is a type of attack where a malicious actor creates multiple fake identities to gain control over a network. Since these fake identities can initiate fake sessions in vehicles, it's an attack type where attention should be paid to identity authentication in vehicles. In Node impersonation attacks, where fake wireless networks can be created, VANET networks with the same name can be formed, and the information transmitted by the vehicle and the driver who wants to initiate a session can be stolen. Therefore, precautions need to be taken against these attacks and connection protocols need to be established. In VANET networks where private messages and safety messages are shared, message suppression attacks which blocks the message that delivered to destination, alteration attacks where messages are modified, and relay attacks where an important message is later retransmitted for a fake alert are types of attacks that need to be taken seriously. Such attacks can lead to forged sessions and locked authentication processes by altering, blocking, and subsequently reusing the information transmitted during authentication. GPS spoofing attacks are also significant threats to VANET systems as they can potentially cripple the system. By providing false GPS information, these attacks can manipulate the perception of vehicle density in certain areas. This manipulation can lead to the locking of RSUs and service-providing servers.

In [5], it is emphasized that some methods have security vulnerabilities, especially the ones that verify the identity of the drivers, by utilising fingerprint and iris scanning which cannot determine the identity of the driver in real time [5]. In addition, RFID token-supported security and key systems are vulnerable to attacks such as theft, duplication, and delay [6]. In addition to these studies, authentication methods that recognize the driver have also been developed by analyzing the driver's

behavior with machine learning algorithms. Machine learning algorithms can create a model from past data or current data, and determine whether the next data is compatible with the past data. In machine learning applications in VANET systems, the similarity between personal parameters of the driver such as hand movements, equipment settings, driving characteristics, and location information of the vehicle are analyzed while driving. Since these parameters are unique, unpredictable, inimitable, and non-replicable, authentication methods based on behavior analysis by machine learning come to the fore [7]. The security of historical data is important in authentication methods to be supported with machine learning algorithms. Since the created model analyzes the similarity between the current data and the historical data, the accuracy of the historical data that is used to create the model will directly affect the result of the authentication. In the studies examined, driver-related data is provided by a central TA (Trusted Authority) or CA (Certificate Authority). This system, based on a centralized structure, is not a secure data storage method. In our study, instead of these structures, the use of blockchain ensures that driver data used for authentication is stored in a secure structure.

In recent years, blockchain technology is one of the subjects studied in VANET systems on data security. Blockchain is a secure database where historical data cannot be deleted or changed, and the order of transactions is preserved. In the structure where each data block is linked to the previous data block, any change in the blocks causes all subsequent blocks to be invalid. In addition, more than %33 of all peer devices in the chain must be attacked at the same time to make modifications in a block. Due to its security, blockchain technology has been applied in the dissemination of sensitive information in VANET systems. In an authentication system that is based on driving behavior analysis with machine learning, the blockchain is also a suitable data storage structure. In authentication studies based on blockchain and machine learning supported driving behavior analysis, hardware assumptions are made for

the system to work and meet security requirements. In [8], cars act as peers in the blockchain, so it is assumed that each car has high computational power and capacity. This high computational power and capacity increases the cost of cars and can complicate feasibility in real life. In [3], RSUs are peers in the blockchain, so they are considered to have high computational power and capacity. Having high computing power and high capacity can make it difficult to meet the investment costs and especially the energy needs in rural areas for RSU. It also increases the amount of energy wasted globally. The assumptions in these studies are accepted to prove that the proposed systems can work in the absence of some constraints. The computing power and capacity required by the hardware of the blockchain structure is a constraint for the systems to be designed. In addition to these constraints, the following issues regarding driver and environmental conditions in real life, should be considered:

- Driver can use different brands of vehicles
- The driver can drive vehicles belonging to different segments
- Driver desire to protect personal information
- Limited hardware investments of manufacturers for vehicles and servers
- Inability to set up equal and high capacity RSUs to the city and rural areas (investment cost, energy constraint)
- Desire of the manufacturers to protect their know-how

The system to be designed for authentication in VANET systems must be sustainable, flexible, accessible, and secure for both manufacturers and drivers. Therefore, the authentication method should have a suitable solution against all the above constraints. In this paper, we propose an authentication system in which the driver can drive vehicles of different brands and segments. To provide this, we also included

the brand, model, and segment data of the vehicle in the driver's driving data. Even if the driver remains the same, the driving response of different segment vehicles may vary. Therefore, filtering the driving data based on vehicle information during authentication ensures a more accurate dataset for authentication. This filtering directly creates a system that allows the driver to use different vehicles. Our study is the first to include different vehicle manufacturers within the same VANET network, enabling secure identity authentication for drivers to use vehicles of different brands. In the study, a new framework has been proposed to achieve this. The framework we present includes unique implementations for vehicle manufacturers to securely share driving history data among each other to provide secure and fast identity verification. The blockchain technology is used to provide trusted data sharing for the car manufacturers. Due to the use of blockchain, feature selection has been applied to the raw driving data to avoid storing high-volume data on the chain, and the dataset obtained through feature selection has been stored on the chain. The datasets obtained through feature selection have been used as training sets for machine learning algorithms used to identify the driver's identity. It has been observed that the machine learning models built using the dataset obtained through feature selection achieve higher accuracy compared to the raw data. Additionally, in the proposed framework, RSUs and vehicles with low hardware power have not been included in the blockchain, thus preventing hardware constraints arising from the blockchain in these components. As a result, with our study, we proposed a new framework that includes different vehicle manufacturers in the VANET environment, provides secure data sharing among manufacturers, avoids storing high-volume raw driving data, and enables faster and accurate secure identity authentication for drivers using vehicles of different brands and segments.

1.2 Background

In this subsection, we will provide background information for blockchain, VANETs, feature selection and machine learning since these are the main blocks of our thesis.

1.2.1 Blockchain

Blockchain is one of the most important data storage structures of the 21st century. Instead of storing data in a single copy like traditional centralized systems, it creates a healthier platform for data security and transaction integrity by maintaining separate copies in distributed systems. Here, in addition to holding data in distributed systems, the way data is recorded ensures that it is immutable, undeletable, and non-replicable, heralding a new era in data storage.

Originally conceptualized as the underlying technology for Bitcoin, a digital cryptocurrency, blockchain has since evolved into a versatile tool with applications spanning finance, supply chain management, healthcare, and voting systems. The mentioned immutability, non-replicability, and undeletability of blockchain enabled its rapid proliferation in sensitive areas such as banking and finance where data security is paramount.

The fundamental principle of blockchain is to create a decentralized and immutable database where transactions are recorded chronologically and transparently. This decentralized structure eliminates single points of failure and reduces the risk of censorship, and manipulation. Additionally, blockchain enhances transparency and accountability by providing all participants with real-time access to a shared ledger, fostering trust and collaboration in multi-party transactions. Each block in the blockchain is cryptographically linked to the previous block, creating a chain of blocks where all data is interconnected. This structure ensures the integrity and security of data, as any attempt to alter a block would require the consensus of all points in the distributed system since it would affect the entire chain. As each added block forms the

blockchain, blocks cannot be deleted or altered. This makes it practically impossible to tamper with the data stored on the blockchain.

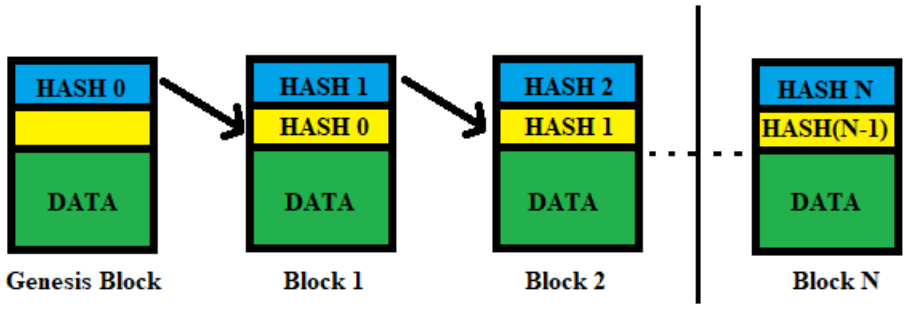

Figure 1: Blockchain Data Structure

In today's world, processes such as data exchange, new record creation, and data deletion typically require approval from intermediaries and central authorities. We can use the example of a car sale to illustrate this. While the ownership of the vehicle and the identity of the new buyer are recorded as data, approval and registration of this transfer are required by agencies such as notary offices. In case of any discrepancies or data deletion, these authorities guarantee the integrity of the transaction through their archives and official seals. Blockchain technology uses consensus mechanisms and cryptographic algorithms to verify and record transactions in data processing. With these methods, the accuracy of each transaction can be verified during and after registration. Therefore, blockchain technology eliminates the need for intermediaries or central authorities.

The decentralized nature and consensus mechanism in blockchain technology, along with immutability and resistance to alteration, are the fundamental aspects that distinguish blockchain. These foundations, built using data blocks, cryptographic hash functions, and peer-to-peer networks, have enabled blockchain to be a rapidly implementable and beneficial data technology in various sectors. Due to the benefits brought by these foundations, blockchain can be easily applied to sectors

such as supply chain and healthcare. Especially in the financial sector, blockchain facilitates peer-to-peer transactions, cross-border payments, smart contracts, and tokenization of assets, making it easier to transfer and manage value. In supply chain management, blockchain records the journey of products from production to consumer, enabling end-to-end traceability and counterfeit prevention. Privacy is crucial in healthcare, and blockchain ensures the integrity and confidentiality of patient data, facilitates data sharing among different doctors and organizations, and enables secure sharing of medical records among stakeholders.

Like any structure, blockchain faces challenges during implementation. Scalability is one of the significant challenges encountered in the blockchain structure. As the number of transactions increases in the blockchain, the response times of the system also increase. Therefore, a balance must be struck between the number of peers, transaction volume, and desired response times. Additionally, since blockchain is a peer-to-peer structure, there is difficulty in secure and easy data sharing between different blockchains. Protocols and standards for these exchange transactions are still in the development stage.

1.2.2 Vanets

Vehicular Ad-Hoc Networks (VANETs) are intelligent systems (ITS) that enable communication between vehicles by establishing dynamic connections among them, thereby improving processes such as traffic efficiency and road safety. VANET systems facilitate proactive measures using vehicle to vehicle (V2V) and vehicle to infrastructure (V2I) communication to optimize transportation and ensure safety, addressing issues arising from increasing urbanization and vehicle density.

At the hardware level, VANETs consist of On-Board Units (OBUs) installed in vehicles and Road Side Units (RSUs) located along roadsides. These units form a communication backbone between vehicles and infrastructure/services, ensuring

seamless communication in dynamic environments. This structure of communication provides robustness, scalability, and compatibility.

VANETs can encompass various networks, including security, traffic management, and entertainment services. Security applications include collision prevention systems and emergency notifications aimed at reducing traffic accidents. Traffic management applications aim to optimize traffic flow, reduce congestion, and enhance efficiency using VANET data.

Additionally, VANETs offer a range of entertainment services aimed at enhancing the driving experience; in VANET systems with internet access, drivers can access entertainment platforms, bank accounts, and specific applications from their vehicles. Driver data is also transmitted within the VANET's entertainment and service layer.

Despite the numerous opportunities offered by VANETs, there are several challenges that need to be overcome to establish and maintain this system. These include network scalability, mobility management, security vulnerabilities, and ensuring Quality of Service (QoS). Below is information structured into subsections to better understand VANETs, including their structure, encountered challenges, applications, and protocols.

1.2.2.1 Architecture

The architecture of VANETs primarily consists of two components: OBUs and RSUs. OBUs can communicate with different vehicles, utilizing sensors and the vehicle's ECU(Electronic Control Unit) hardware [9] Therefore, they have extensions in both hardware and software to support various communication protocols. RSUs, on the other hand, have stronger hardware and continuous power supply, positioned along roadsides. These devices enable V2V and V2I communication by forming a Wi-Fi network with mobile OBUs [10]. In figure 2 , structure of a simple VANET network illusturated where both V2V and V2I architectures coexist within the same network.

In VANET systems, RSUs facilitate the transition between wired and wireless communication. RSUs integrate vehicles, which are mobile systems, into the network using wireless communication protocols. They also incorporate fixed location service providers into the system using either wired or wireless communication protocols. Trusted Authorities (TA), Certificate Authorities (CA), and cloud service providers also communicate with vehicles through this network. Additionally, vehicles, being mobile systems, can communicate with each other and with local peripherals using other short-range communication protocols embedded in their hardware, apart from RSUs.

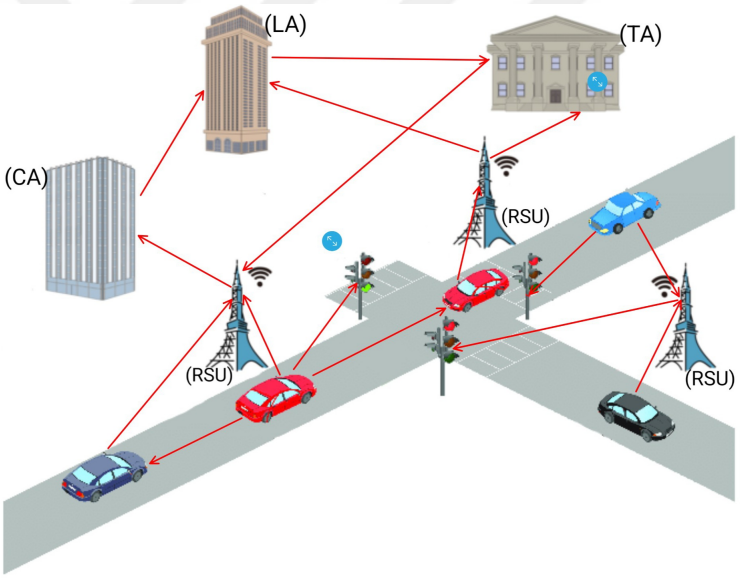

Figure 2: VANET Structure

1.2.2.2 Communication Protocols

VANETs employ various communication protocols to enable efficient data exchange among vehicles and infrastructure components. The IEEE 802.11p/WAVE (Wireless Access in Vehicular Environments) standard is widely adopted for VANET communication due to its suitability for high-speed mobility and short-range communication requirements [11]. This standard operates in the 5.9 GHz Dedicated Short Range

Communications (DSRC) band, providing low-latency and reliable communication for safety-critical applications. Additionally, cellular networks, including emerging 5G technologies, are being explored to complement IEEE 802.11p/WAVE by providing wider coverage and connectivity beyond the range of RSUs [12].

1.2.2.3 Applications

VANETs are network architectures that enable information exchange between mobile systems and infrastructure systems. Communication between infrastructure services and vehicles, as well as surrounding systems, facilitates the integration of applications that enhance road safety, traffic efficiency, and serve end-users.

Real-time information about road conditions, coupled with data on congestion and accidents from vehicles in traffic, can be used to inform approaching vehicles and suggest alternative routes. Through this method, travel time, fuel efficiency, and most importantly, traffic safety can be improved, enabling efficient traffic management and optimization. The study in [13] serves as an example of this approach.

Vehicles can exchange data with each other through VANET systems. OBUs, which are fundamental components of VANET systems, can share information such as the vehicle's direction, speed, acceleration, and route data with other vehicles using car sensors. Vehicles can use this information to perform calculations and determine potential collision points with other vehicles. Thus, collision prevention systems can be developed using VANET networks, as illustrated by the example study referenced in [14].

In the event of an accident, vehicles can report their location and status to emergency services through VANET systems. After an accident, drivers can share their health information by retrieving data from their phones. Moreover, in shopping centers and urban areas, VANET systems can transmit suitable parking spots based on

the size of the vehicle. Additionally, an example of an application studied in [15] involves displaying advertisements on roadside billboards tailored to attract the driver's interest during their commute.

1.2.2.4 Challenges And Research Trends

Despite the potential applications and benefits of VANETs, there are challenges that hinder their widespread adoption and usage. One such challenge is the limitation of network scalability due to restricted bandwidth in urban areas. Additionally, another constraint arises from the dynamic network topology during transitions between RSUs (Roadside Units) in scenarios where vehicles have very high speeds, which can limit healthy data exchange due to connection establishment and disruption times. The absence of reliable data exchange leads to network delays, packet losses, and other issues, thus posing challenges related to Reliability and Quality of Service (QoS). Security and privacy concerns are also critical in VANETs, and also are vulnerable to various security threats, including GPS spoofing, eavesdropping, jamming, sybils, alteration, copying, node impersonation, message suppression, relay, alteration, DDoS and malware attacks. These are common attacks seen in VANETs. GPS spoofing attacks involve altering the GPS information transmitted by a vehicle to local authorities or surrounding vehicles, deceiving authorities with fake traffic congestion or location information, and potentially causing accidents. Eavesdropping is listening a data transmitted between communicating devices. Although not an active attack, particularly in VANET networks, eavesdropping on information sharing among vehicles of the same manufacturer or listening to the driver's private information can be exploited for malicious purposes. Jamming attack is the interruption of communication signals in the environment with electromagnetic waves which has same frequency with communication channel. Although this attack can be used for security purposes in military operations, it can also be utilized by malicious individuals to disrupt the

communication of a device or vehicle with the intention of causing harm. In such cases, the vehicle needs to quickly detect and respond to this attack by implementing security measures such as locking or requesting emergency assistance from a different frequency. A Sybil attack aims for an attacker to take control of a network by creating multiple fake identities. In VANET networks, this can involve creating fake vehicles or fake roadside units to cause effects such as traffic congestion, accidents, or slowdowns in local servers. Alteration attack is the modification of data transmitted during communication. If this attack occurs in an unencrypted communication environment, it can lead to tampering. For example, when the information of a person logging into a vehicle is sent from the server, it can be altered, and the person may not be able to log in. A copying attack is an attack where a message transmitted during communication is copied. Although it's a passive attack, sharing or reusing the copied sensitive data can lead to harmful actions. Node impersonation attacks involve impersonating a node to steal information from or gain control over devices communicating with that node. Especially in VANET systems, malicious networks with the same wireless network name as the broadcasted network can capture sensitive information from vehicles when they connect. Message suppression attacks directly affect communication by preventing the transmission of data sent or expected by communicating devices. The difference between this attack and a jamming attack is that in message suppression, the communication channel exists but the messages transmitted over the channel do not reach their destination. These attacks can lead to the non-delivery of important information such as emergency messages and hazard alerts in VANET systems. Relay attacks involve capturing messages during communication and retransmitting them after a certain period to deceive the receiver or prompt a repeat action. Particularly in bank transfers such as electronic fund transfers (EFT), repeating transactions can lead to significant losses. Similarly, in VANET systems, relaying data such as hazard messages and toll payments can result

in harm if transmitted repeatedly. These attacks and examples are types of attacks encountered in VANETs and can affect servers, vehicles, and roadside units. Ensuring the security and privacy of communication among vehicles and with infrastructure elements is essential to prevent unauthorized access, data manipulation, and privacy breaches. Efforts are underway in VANET systems to overcome these challenges by developing more reliable protocols that provide fast connections and creating network structures capable of handling heavy data traffic. Research focuses on V2V and V2I communications in urban environments, traffic management, congestion reduction, authentication systems, and communication protocols. Particularly, machine learning algorithms and blockchain technology are being integrated to explore new approaches and create more resilient systems.

1.2.3 Machine Learning

Machine learning is a subset of artificial intelligence technology. It focuses on developing models that learn the relationships in data. By learning these relationships mathematically, the trained models make estimations and predictions about new data. Development of a model with machine learning consists of different techniques and requires modifications depending on applications. The important point in determining the development techniques is to understand the data that given to the algorithm to learn the relationships. A fully labeled dataset, an unlabeled dataset, or a partially labeled dataset are learned using different machine learning algorithms and methods. Depending on the learning type, machine learning is examined under four categories: supervised learning, unsupervised learning, semi-supervised learning, and reinforcement learning. Supervised learning deals with labeled data, where both the inputs and outputs are known, and the supervised machine learning algorithms find the relationship between inputs and outputs. In unsupervised learning, data is unlabeled. It is more complicated than supervised learning, and machine learning algorithms

find hidden clusters and patterns to label them. In semi-supervised learning, data includes both labeled and unlabeled data. Machine learning algorithms find patterns and clusters and also develop the model with labeled data. In reinforcement learning, machine learning algorithms take an action, produce outputs, and observe the environment's reaction. Through this trial-and-error method, the model improves and achieves optimal results.

The machine learning algorithm could be applied on various datasets for learning the data. In authentication data, the aim is to verify the person if he is really the guy who he claimed. In identity verification for VANETs, machine learning algorithms are trained with raw data gathered from car sensors while drivers are driving. The raw data includes a set of driving data of drivers. Because of the drivers are known while driving, the raw data is labeled. Machine learning algorithms use this data to learn and test themselves to understand the different driving behaviors of drivers. After learning, the model can determine the driver to whom new unlabeled data belongs. So, supervised learning methods which deals with labelled data could apply. In supervised learning methods, K-Nearest Neighbor, Decision Tree, Multi Linear Regression, Linear Regression, and Random Forest are common machine learning methods applied to labeled data.

kNN is a powerful supervised learning algorithm that used in regression and classification of data. The main idea of the algorithm is labelling the new data to class which the majority of its surrounding n neighbors belong. In the performance of the algorithm, n defines the selected number of neighbors to consider in classification. The value of n should be optimal because the class to which new data is assigned depends on the value of n. For example, if you choose n to be two and label the data as class one based on these neighbors, but if you choose n to be five, the majority might change, resulting in the data being labeled as class two.

Decision tree is also a powerful learning algorithm for supervised learning. It

defines branches to question the feature relevance of data with each subbranch and moves to another subbranch over the more relevant one, and finally, the branch ends with a leaf which determines the class of the data. The algorithm is named decision tree because its decision-making mechanism resembles the branches of a tree when determining the class of the data.

Linear regression and Multi Linear Regression have similar learning techniques. They are also supervised learning methods and could applied in continues data to make predictions about results. It improves the model by learning parameters and result in numerical datasets. Difference between Multi Linear and Linear Regression is variable number that is considered. In Linear regression independent variable x_1 fits to calculate dependent variable y , but in Multi Linear Regression two or more independent variables x_1, $x_2 ... x_n$ fits to calculate y. While it is not generally used for classification, it could be applied in binary classification. In our dataset, drivers are labeled with driver IDs, and all features are numerical. We could try to find the relationship between driving features and the target value. The target value is a column in a dataset which will determine with analysing and fitting of other columns in data by machine learning. In this scenario our target column is identity of drivers. To apply Multiple Linear Regression (MLR) to our dataset, we should change the target values which is driver identity to zeros and ones. If we label the identity of driver who he is claimed to be as ones in feature dataset and the other driver's identity as zeros in feature dataset, then we could apply binary classification. Because the target value which is driver identity transforms zero and ones. The accuracy result will provide information about whether the model fits the dataset or not.

The Random Forest is an improved version of a single decision tree. It includes more than one decision tree and assigns the data to each of them for classification. After all the trees reach a solution, the Random Forest calculates the majority class determined by the trees. The majority class is then defined as the data's class.

1.2.4 Feature Selection

Feature selection is a method used to create a subset of features or variables within an original dataset. This technique aims to generate a smaller dataset from a large-volume dataset. The new dataset possesses superior characteristics such as dimensionality reduction, increased interpretability, reduced computational time, and enhanced model performance by removing redundant or similar data. Feature selection methods, which can be categorized under three main headings as Wrapper methods, Embedded methods and Filter methods create a more suitable dataset for specific applications by selecting features using different methods.

Wrapper methods train a predictive model to select features from the dataset and choose the features based on their impact on the model's performance. Recursive Feature Elimination (RFE) is a method that applies wrapper method model. Embedded methods evaluate feature selection as part of the model training process. During model training, if the impact of a feature is low, it is removed. Lasso regression is example for embedded methods. Filtering method evaluates the data statistically, hence the methods under the filtering method are based on a mathematical foundation. The Pearson Correlation Coefficient method, Spearman Rank Correlation method, Mutual Information method and Chi-Squared test method is well known as filtering methods. Correlation-based Feature Selection (CFS) is also filtering method for feature selection and evaluates the correlations of the data and identifies features with low inter-correlation.

CFS is chosen for our study because it is more suitable for our dataset due to two key features that distinguish it from other feature selection methods. When performing feature selection with CFS, it does not only consider the correlation of the selected feature with the target feature but also examines the correlation between the selected feature and other features that define the target feature. Additionally, CFS selects features with low correlation with each other, meaning it chooses features

whose changes are not dependent on each other. This highlights unique features in the data. For example, when a driver suddenly accelerates, there is a natural simultaneous increase in throttle opening and speed data, resulting in a high correlation between these two features. Thus, using them to characterize the driver would provide little valuable information. On the other hand, throttle information and sudden changes in steering angle are independent of each other, making them more valuable as features for characterizing the driver. Therefore, by considering both the correlation with the target feature and the independence from other features, CFS ensures that the selected features provide meaningful and unique information for our dataset.

CHAPTER II

RELATED WORKS

The structure we propose for the driver authentication system includes blockchain, cryptography, and machine learning implementation for VANETs. The studies in the literature on related topics are compiled under the following subsections.

2.1 Machine Learning Based Driver Identification

Behaviours and habits are the most important features that define us. Hand movements, speech speed, residing area, travelling zones and reaction time are parameters that may differ from person to person. Although these parameters seems similar for two people, there is a unique change in parameters for each person.

Machine learning algorithms analysis the data and apply learning techniques to understand the change in parameters in each data for classifying them. With the rapid development of vehicle technology, cars are equipped with sensors that gather data while driving to help the driver in driving. The driving style of the driver also can be observed in changes in sensor values. This classification methods also provide us to identify the driver by analysing the change in driving style. At this point, machine learning algorithms could be applied in cars to identify the driver.

Nowadays, cars get smart technology and have more special information about the driver. This special information also could include bank accounts, passwords, payment numbers, and web accounts in addition to driving setups. Since this data contains personal information, security methods should be implemented on vehicles when accessing the data. Authentication is the most crucial of these methods. Authentication could be questioned in log-in and the others sessions. But in accessing sensitive data, logging in is an important step so in our work we focus on that.

Some cars also have different sensors and systems for authentication. Fingerprint and Iris scanning is an authentication method provided with sensors, and they are not real-time verification methods. For this reason in [5], a new authentication method has been developed by combining data from the vehicle with CNN (Convolutional Neural Network) and SVDD (Support Vector Data Description) algorithms . In this article, the authors designs a system that runs locally. Each car has its decision program and past driving data for the driver. In our scheme, cars share driving data globally and it increases the reliability of training data.

In [16], the authors proposes an authentication method that identifies the drivers from the driving data of the car. With the help of Principal Component Analysis, variables with high variability were determined to reduce analyzing time. Then, by analyzing the data with the Extremely Randomized Tree algorithm, authentication is performed. The authors also proposes a new classifier from a probability distribution to identify the suspicious car in VANET. In addition, the authors used a dataset of past driving data of all drivers to train the machine learning algorithm. In our proposed method, there are two different data sets that belong to single driver that contain sensor data and historical driving parameters. Since the machine learning algorithm applied in [16] creates a model by examining the data of all drivers, the computation time to create the model also increases as the number of drivers increases. Additionally, when a new driver is added to the system, the entire model needs to be recalculated. In our proposed system, each driver is evaluated only with their own historical data, hence the computation time is lower, and adding a new driver to the system does not affect the model of other drivers.

In study [17], smartphone and vehicle sensors were collected using CNN and RNN/LSTM to develop a driver authentication method. Unlike other studies, cross-validation was applied in this study to present a reproducible method against realistic data. In this work, the authors propose a classification algorithm that runs

locally. On the other hand, they used data that was collected on the same path and in the same car. In our scheme, data could be collected from cars that the driver used randomly but in the identification phase, we just consider the driving behavior parameter of the driver in similar cars which has the same power parameters. Cars with different power values can have different acceleration, deceleration, and sensor values during driving, regardless of the driver. Therefore, while a model created based on the driving behavior of cars with the same power values can recognize the driver, a model created with values from cars with different power balances leads to under fitting.

In [18], GPS data was used to verify the identity of the driver in the vehicles. With this GPS data, the location information of the points where the driver went during the trip was taken and a feature of the driver was obtained from this data. These features were analyzed in the random forest classification to determine whether the driver was person X. In our work, we will answer the same question the driver is X or not by identifying the driver by machine learning algorithm but we will use car sensor data because the GPS data of the driver could be the same as with family individuals.

In the study in [19] a method is proposed for the authentication of taxi drivers. In this method, the driver worked on detecting the identity of the person from hand movements and swiping movements while using the phone, authors assumed that the driver definitely contact with the phone while using the application on the phone when the journey was booked. In our work, we design our system for everyone who drives, it could be a taxi driver or a truck driver. To provide that we focused on data that is common for all driver profiles so we analyse the driving behavior from car sensors.

One of the most important differences between our study from other studies is that sensor data is not used as comparison data. Sensor data is permanently deleted

after the model is created. Therefore, there is no access to historical sensor data. In the studies in the literature, the incoming sensor data is used for comparison, but in our study, it is only used to create the model. We verify the identity of the driver by examining whether the current model and the previous models are similar. Thus, the size of the data we keep permanently is reduced. Additionally, a model is created at the end of each drive, which means there is one meaningful data point for each drive. Creating a new model to compare the current model with past models can lead to under fitting as the new model will have fewer samples. To overcome this, we can increase the number of sampled parameters by creating a model from data at different time intervals during the drive.

2.2 Cryptography in VANETs

Many encryption and data storage methods are used in VANET systems to increase security and expand their scope. In these studies, the interpretation of encryption and data storage methods with environmental data and hardware sensors has revealed different approaches to increase security.

In [20], which is one of the studies on security in communication, a method based on a 4-stage fixed key infrastructure has been proposed to prevent Sybil attacks. A Sybil attack is a type of attack where the attacker creates multiple fake identities in a network, allowing them to control the network with these identities, deceive a server, or slow down the server's operation. In this method, authentication, data integrity, Non-repudiation, and privacy in security issues are provided. At the end of the study, it was observed that the execution time of the method was low due to the CA doing the load, therefore it was stated that the proposed method was the most effective in Sybil attacks. In addition, it was observed that the number of messages rather than the number of cars was effective in the delay in detecting the attack. In this work, the system gets slow in detecting the attacks when the number of messages

increases. In real life millions of cars communicate at the same time so, this method could not fit to our structure, we desire to design a system which is robust for a dense communication network.

In [21], a method targeting 5 different points (an authentication mechanism, privacy, anonymity, private communication, and information security) is proposed. At the end of the study, it is stated that the proposed method has a lower encryption/decryption time compared to the referenced studies and meets the confidentiality, authentication, non-repudiation, conditional anonymity, and conditional traceability features in terms of security analysis.

The method in [22] is proposed to detect DDoS attacks, which are among the most common attacks affecting communication. In the study, it is focused on detecting the beginning of the DDoS attack according to the rate of change in the position information and messaging frequency coming from the vehicle's OBU. At the end of the study, a method that can be used against DDoS attacks has been obtained. We do not save raw data in our system, which ensures that the communication frequency between the server and the vehicle is very low. Therefore, any potential DDoS attack can be quickly detected. In addition, since the raw data has a very complex structure, any content alteration attack on this data may not be detected. Any change in the data can have a permanent impact on the identification of the driver. Therefore, this data needs to be isolated from attack-prone structures such as the network.

In [23], the authors proposed ACAFP, modified RSA algorithm, which uses four prime number to generate private and public keys for encryption. In this study, four prime number which denoted as o, k, r, s have same length, are chosen and n is calculated as n=o*k*r*s. After n is found, f(n) is calculated f(n)=(o-1)*(k-1)*(r-1)*(s-1). As in same in RSA, p is choosen between 1<p <f(n) where f(n) not divisible by p. As last to find public and private keys d is calculated with p(mod f(n)). As a result of this work, in encyrption and decyription ACAFP algorithm

needs less computational power and memory than the RSA algorithm. On the other hand with the same computational power, ACAFP is faster than RSA according to key generation time. In our work, we need a simple, fast and reliable encryption algorithm and ACAFP provides this. Because of that, we applied this method as our encryption algorithm.

2.3 Blockchain for VANETs

Since blockchain provides integrity, and non-repudiation it is widely utilised in VANETs. By using blockchain structure, in [24], key exchange, key generation, and session confirmations in communication are provided by the ECC Diffie-Hellman algorithm and these key data are stored in the blockchain. In our work, we will keep drivers driving model parameters in the blockchain structure.

In [3], where events are recorded in the blockchain structure, a distributed, traceable communication structure is proposed. In this study, a model was created by making some hardware assumptions (computing power, storage capacity). Once the vehicle is authenticated by a Cloud, its identity is authenticated. After this verification, the data sent by the vehicle is stored in the blockchain. In this work cars are members of a blockchain. This structure is not fit for our proposed method, since we want to build a sustainable system and keep the cars out of the blockchain. Instead, the servers of the manufacturers create a blockchain structure among themselves.

In [25], a blockchain-based access control mechanism scheme is proposed. With this study, a scheme that can protect against seven different attack types has been created. In [8], cars are considered member of the blockchain, so their computational power is assumed high. In this system, the message sent by the car is verified by %51 of the cars in the group of which the car is a member and added to the blockchain as a ledger, and broadcast to other vehicles. Thus, an effective method for the dissemination of critical messages has been proposed. In our scheme vehicles are not a

part of the blockchain, so they can operate our system's machine learning algorithms without requiring additional computing power. Furthermore, as the blockchain's capacity needs increase over time, there is no need to add extra capacity to vehicles as they are not on the blockchain. Additionally, the operating system on vehicles is not as secure as servers from a security perspective, so it is safer to store sensitive data on peer-to-peer nodes, such as servers, which are more secure against copy attacks which is refer to unauthorized duplication or replication of sensitive data.

In [26], instead of cars, RSUs (Road Side Units) are blockchain peers. In this study, the message sent by a vehicle is transmitted by the RSU to the vehicle which has the highest reliability score in the group, and the accuracy of the message is confirmed by this vehicle. Then, the message is broadcast and the reliability score of the vehicle that sent the message is recalculated. In our scheme Road Side Units just provides a connection between cars and servers, and because of that RSUs are not a member of the blockchain. Since RSUs are roadside structures, it is difficult or limited to provide energy to them. Even though they usually get their energy from a fixed power line, they can stay powered by solar energy or batteries in some regions. Therefore, the low computational load will directly affect the lifetime and investment costs of these systems. In our scheme, the computational load of the system is on the server that builds and manages by the manufacturer.

In [27], pseudo names used by vehicles during communication are stored in two different blockchain as current and past. The pseudo name produced by the car for authentication must not be the same as one of the active vehicles and must not be the same as a past pseudo name. Our authentication method use analysing algorithm to authenticate the driver so it is more complex and powerful way then stamping with pseudo names.

Finally, in [28], the consensus mechanism of the blockchain and Q learning-based action selection strategy are used. This selection strategy is aimed to store the data

by choosing the optimum number of actions to help the behavior of the OBU. The main purpose of this study is to reduce the attack potential in communication. In our work, we kept the model parameters of the machine learning algorithm in the blockchain structure so communication frequency is low.

In our proposed work, taking into account all the studies conducted, we proposed a new structure that provides security against attacks on both the blockchain and VANET systems. Here are the advantages that the system we propose offers, which are different from the ones presented in the literature:

- The capacity requirement is lower compared to other systems because sensor data is not recorded permanently.
- The driver can use different brands and models of vehicles, and the system has access to the model parameters of the driving history of all brands and models in the blockchain to verify the driver's identity.
- Cars and RSUs do not need high computing power and capacity since they are not members of the blockchain.
- No component in the VANET network is a peer of the blockchain. Therefore, two separate private networks have been established: one for the VANET network and one for the Blockchain network. The independence of the blockchain and the VANET networks allows them to implement network-specific settings and security approaches for each network.
- The need for additional computing power and capacity due to an increase in the number of drivers and registered data in the system will be more flexible in the blockchain due to the presence of only servers.
- In our system, the computation load for block calculation and communication with vehicles is on the manufacturers' own servers. The block calculation is

performed by the manufacturer's blockchain member server.

- We train our machine learning algorithm with drivers' own historical data instead of training and classifying with raw data for all drivers, which allows us no need re-evaluate all data for newly added drivers in the system.

CHAPTER III

PROPOSED WORK: MALBUAV

In this section, we have described the steps of the proposed structure called MALBUAV. Section divided into five stages which are car-server communication , vehicle initialization, data request, authentication, and blockchain data record.

3.1 Architectural Framework

The proposed MALBUAV is a framework designed for authentication purposes in VANETs, where there are multiple vehicle manufacturers and drivers can use different vehicles within the network. The system model of the proposed work is illustrated in Figure 3. In our model, we divide the structure to three different environment (as car level, VANET level and blockchain level) to provide clear definition of components communication borders. The vehicle environment level consists of OBU, ECU, and gyro sensors. In this level, devices communicate with each other via the CANBUS line, which is a common protocol in cars. Each device has a different purpose and authority in managing driving functions. The ECU has authority over commanding driving tools of the car such as the engine, gearbox, and safety systems. The OBU is a device that has a screen and interacts with the driver directly. It enables the driver to control driving features and car accessories via communication with the ECU. In the car environment, only the OBU communicates with the VANET environment level via wireless communication. The VANET level considers cars, RSUs, and manufacturer servers. In Figure 3, the model we have drawn to illustrate the proposed work, vehicles of different colors represent different brands in VANET. These vehicles communicate with their own manufacturer servers via wireless connection through

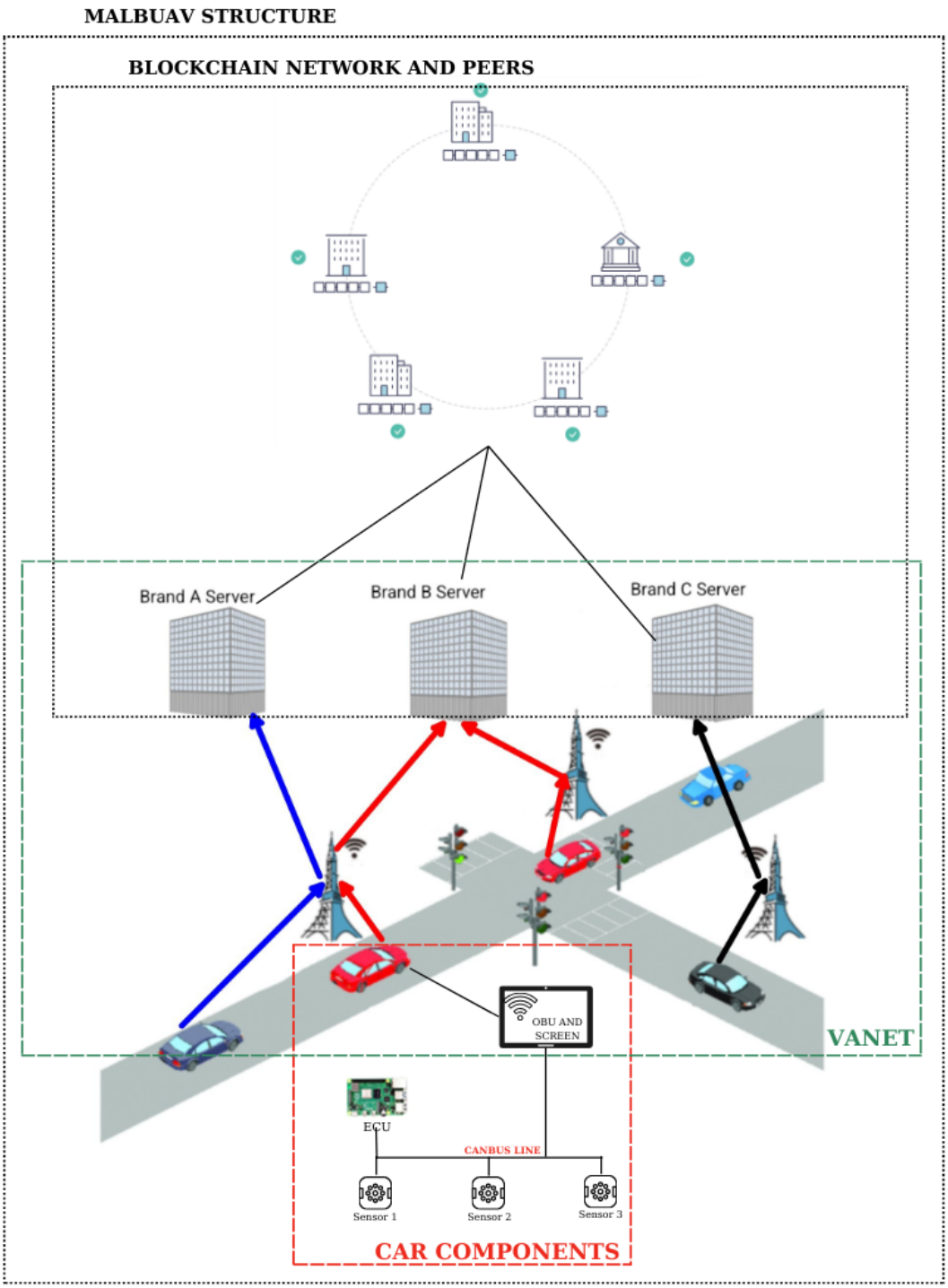

Figure 3: MALBUAV Components Structure

any nearby RSU. In our study, RSUs are programmed solely to provide communication between vehicles, which are mobile devices, and fixed stations, that are server, utilizing all hardware resources to sustain this communication. The blockchain level, includes a blockchain and manufacturer servers. In the blockchain level, a chain is created with peers belonging to each manufacturers. However, it should be clear that the manufacturer server, which communicates with the VANET level, and the manufacturer peer, which is part of the chain, are different servers. In this level, the manufacturer server communicates with the vehicle and evaluates the requests coming from the vehicle. If past driving data is requested, the manufacturer server sends a request to the blockchain peer belonging to the manufacturer, along with the vehicle and driver information. The blockchain peers filter the request based on the information sent by the manufacturer server and transmit it back to the manufacturer server. The framework we propose enables different dynamic -levels to interact seamlessly with each other. Particularly, VANET and blockchain levels have their own communication and network settings tailored to their dynamics. Facilitating communication between these levels through an intermediary component allows them to maintain their individual dynamics and avoids constraints imposed by the unified structure. For instance, if low-resource RSUs and OBUs were included in the blockchain level, they would require high computational and capacity power, which is challenging to provide in reality. In the sections below, communication protocols between the components in this structure and the details of these protocols are explained.

3.1.1 Car and Manufacturer Server Key Exchange Protocol

Our system's client, which is a car, receives all information from a manufacturer server. After the car is started, it connects to the manufacturer server via an RSU to retrieve the necessary information. The RSU just provide a connection point for cars. After connection, cars directly communicate with manufacturer servers. Upon this

connection, the car and the manufacturer server starts a protocol that depicted in Figure 4 for secure communication. In a secure communication protocol, the message should be unrepeated, unchanged,confidential and agents are authenticated.

Our key exchange protocol in Figure 4, is based on the Diffie-Hellman algorithm. In the protocol, nonce values are used to detect message refreshness, and ID information is used to authenticate the agents. In addition, signatures are used in messages to verify that the message is not altered. We applied the ACAPF method described in [23] to generate asymmetric keys for the signaturing. ACAPF uses four small prime numbers when generating asymmetric key pairs, which results in faster encryption and requires less computational power compared to RSA. This method is more suitable for devices with limited computational power, such as OBU.

Proposed key exchange protocol is initiated by the car. The car creates an asymmetric key pair with ACAPF for signaturing. Then, the car selects a random secret integer a and calculates the value R_c using g^a mod(p). The value R_c is the result of the modulus calculation in the Diffie-Hellman algorithm and g is a primitive root in the group of integers modulus p which is predefined. Subsequently, the vehicle selects a nonce value N_1 to freshness and signs the chosen nonce value N_1 and R_c with the asymmetric private key PR_c. The car sends a message to the manufacturer server consisting of its own id ID_c, public key PK_c, R_c, N_1, and $Sign_c(R_c, N_1)$.

The manufacturer server verifies the signature using the public key PK_c from the incoming message along with the variables R_c and N_1. Then it verifies the identity of the vehicle using ID_c. Here, by verifying the signature, it is confirmed that the message is not altered, and by checking the ID, it is confirmed that the message is sent by the car. If the verifications are successful, it proceeds with the protocol. Also, in the Diffie-Hellman key exchange algorithm, the intruder's knowledge of R_c does not reveal any information about the integer a that the car keeps hidden, thus preserving privacy.

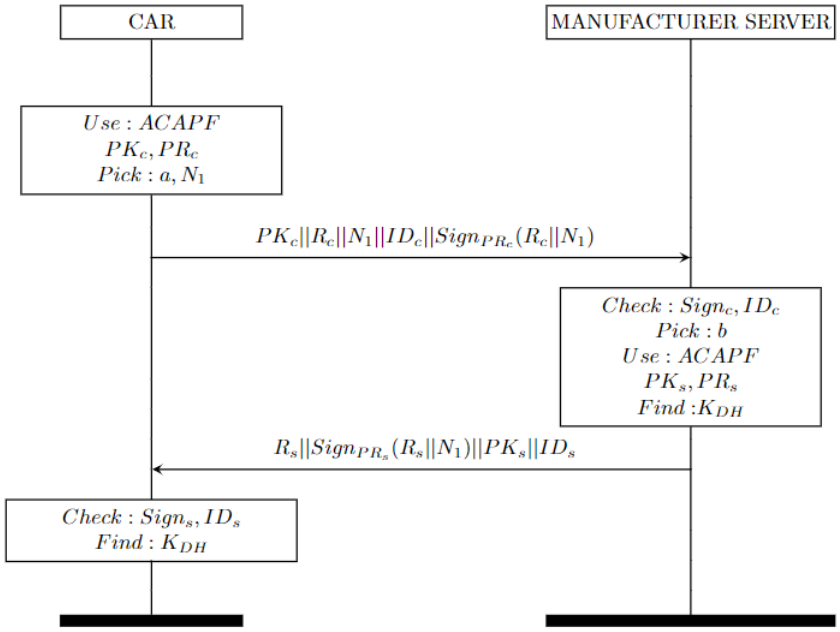

Figure 4: Key Exchange Protocol Sequence

After message and agent authentication, the manufacturer server creates an asymmetric key pair PK_s and PR_s using the ACAPF algorithm then chooses random integer b and calculates the value of R_s with g^bmod(p). The manufacturer server signs the values R_s and N_1 using its generated private key PR_s. The message that is sent to the car includes the public key PK_s, the signature $Sign_s(R_s\ ,\ N_1)$, the value of R_s, and the manufacturer server id ID_s. After receiving the message, the car checks signature with PK_s by using R_s and the locally stored N_1 value. Additionally, since the N_1 value is included in the signature, it ensures that the message is not replayed. After authenticating the server with ID_s is done succesfully, the car calculates secret symmetric key K_{DH} by using R_c and the received R_s value. On the manufacturer server side, since it also knows the values of R_s and the received R_c, it calculates the secret symmetric key K_{DH} as well. At the end of this protocol, the vehicle and manufacturer server securely establish a shared symmetric key K_{DH} , and they encrypt data in communication using this key for secrecy.

The key exchange protocol in 5 tested in AVISPA tool for ensuring the security of

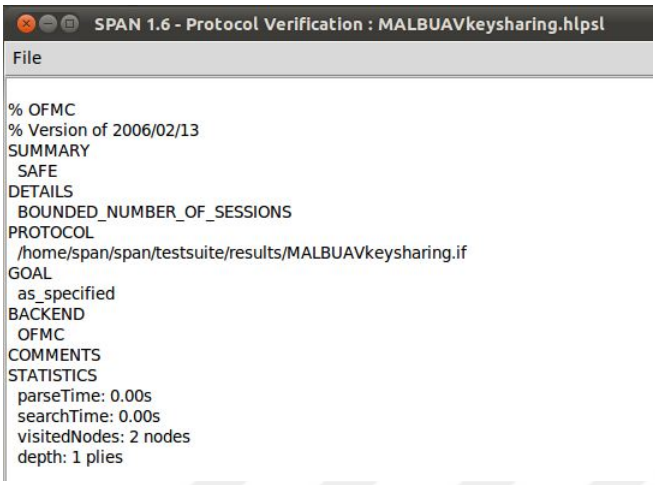

Figure 5: AVISPA Tool Key Exchange Results

the protocol. The analysing results of the AVISPA tool is shown in Figure 5. AVISPA (Automated Validation of Internet Security Protocols and Applications) is a tool for automated analysis of security protocols and applications [29].

3.1.2 Vehicle Initialization

In the recommended system, the On-Board Unit (OBU) is a component that directly communicates with the driver. The OBU has a screen to establish a more interactive communication with the driver. After the key exchange protocol is completed and secret keys are generated for secure communication, the OBU activates its screen and informs the driver that they can initiate the vehicle starting process. The screen remains inactive until the key exchange is completed and secure keys are established, so only after this stage, the driver can start the vehicle. This procedure is for performing preliminary checks before enabling the vehicle's driving functions and verifying whether the driver has the legal authorization to operate the vehicle. The Sequence diagram for this process is depicted in Figure 6.

The procedure begins with the driver's request to activate the vehicle's driving. After entering the activation information from the vehicle's screen, the OBU requests the driver to input the driver's license ID. Once the driver enters the ID, the OBU,

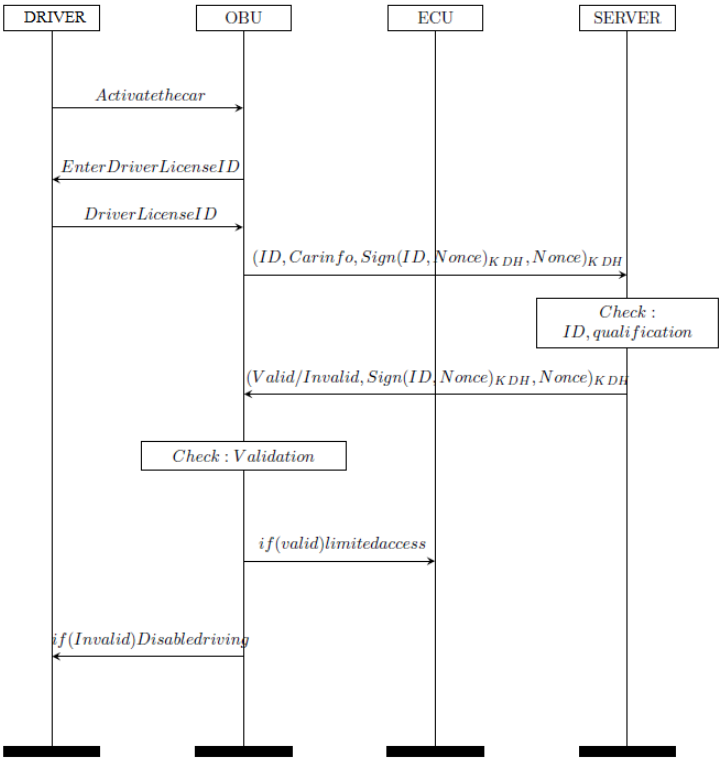

Figure 6: Driving Initializing Sequence

along with the driver's license ID, sends the vehicle's segment information to the manufacturer's server. By sending this data, it also adds the car's signature and a nonce for the freshness of the message, encrypting the message with the secret key K_{DH} to secure communication. The AVISPA result of this protocol is shown in Figure 7 . The manufacturer's server checks signature, nonce and the existence of the driver's license ID then determines whether the driver is authorized to drive the vehicle within that segment based on the vehicle's segment information. If the ID cannot be found or if it is determined that the driver's license cannot be allowed for the vehicles in that segment, the manufacturer's server sends a "no valid permissions" message to the OBU. In this case, the OBU alerts the driver and rejects the request to start driving.

If the driver's ID is found in the database and they have authorization to drive the vehicle, the manufacturer's server sends a "permissions valid" message to the OBU.

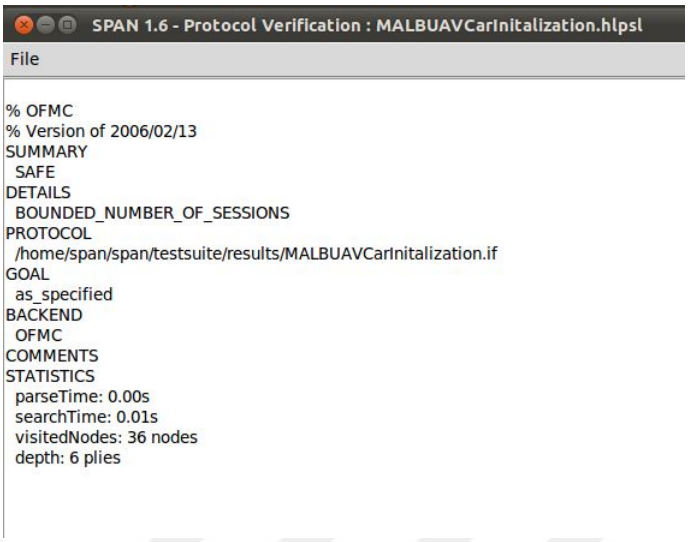

Figure 7: AVISPA Tool Car Initialization Results

In this case, the OBU commands the vehicle's ECU to start driving functions with limited features (maximum speed and maximum range) for authentication.

3.1.3 Data Request

When the vehicle is started, and driving functions with limited features are activated, the authentication process has also begun. The authentication process is a step used for identifying and authorizing the driver of the vehicle. The recommended authentication structure verifies the driver's identity by comparing their past driving behaviors with their current driving behaviors. Therefore, to perform this comparison, the OBU requires data that includes the driver's past driving behaviors from the manufacturer server. Therefore, in this data request section, we explain how the OBU requests the data containing the driver's past driving behaviors and how this data is provided. Figure 8 illustrates the sequence diagram.

For a reliable identity verification method, the data used for comparison must be accurate and appropriate. In the proposed authentication structure, the OBU sends a data request to the manufacturer server, including the driver's license ID along with the vehicle's brand, segment, and model information, to filter the correct comparison

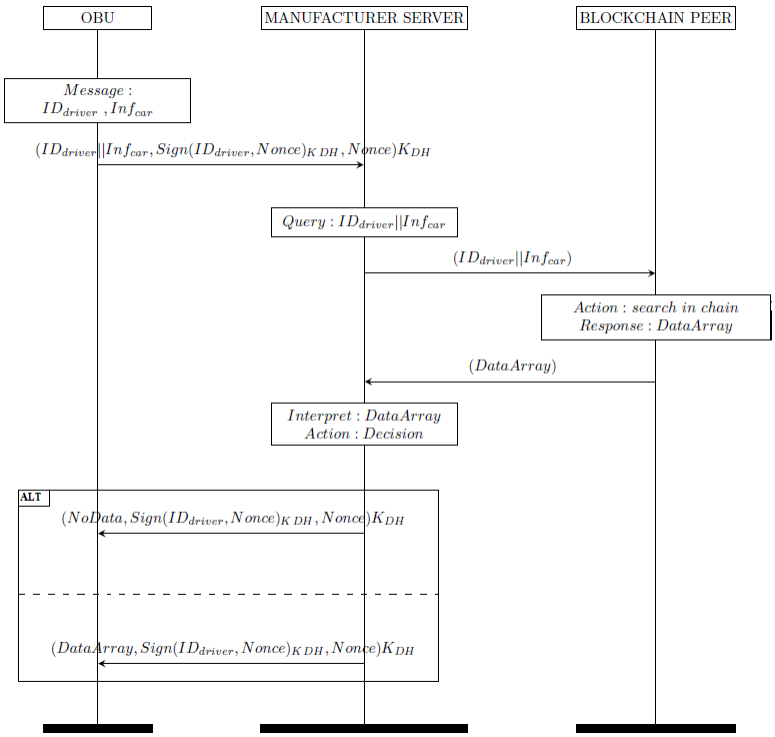

Figure 8: Data Request Sequence

data. The request message also includes signature and nonce to provide security of communication. The AVISPA result of this protocol is shown in Figure 9. When the manufacturer's server receives the data request from the vehicle, it creates a query using the information in the request. The communication between the manufacturer server and blockchain is secured by encryption algorithms that blockchain used. It then sends this query to the peer of the manufacturer's server in the blockchain, requesting the data. The blockchain peer filters the data with the given information and obtains a data set. The dataset includes past driving data of three random driver and past driving data of current driver. This data set is then relayed back to the manufacturer's server.

The manufacturer's server interprets the content of the data array sent by the peer based on specific criteria, such as the number of examples and their similarity. As a result of this interpretation, the manufacturer's server responds to the OBU with one

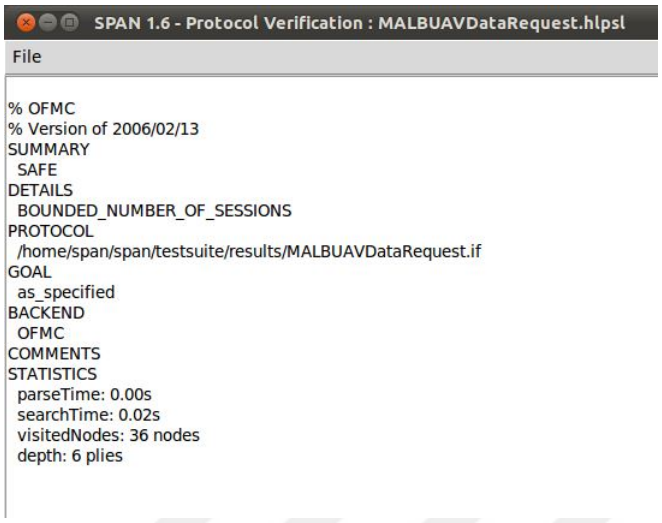

Figure 9: AVISPA Tool Data Request Results

of four possible decisions as an answer.

- If the data array is empty, it indicates that the driver is either someone who has recently obtained a new driver license or has never driven in this segment before, hence no data is available. In this case, as the driver cannot be identified, the manufacturer's server responds to the OBU with the information "no valid data" as the answer.

- If the data array is not empty, then the number of samples in the array is checked. If the number of samples is insufficient for comparison, the manufacturer's server responds to the OBU with the information "no valid data."

- If the data array is not empty, and the number of samples in the array is sufficient, the next step is to check the number of samples for the same brand and model. If the count is sufficient, the manufacturer's server responds to the OBU with the "valid data" information and sends the dataset containing only the samples from that the same brand and model to the OBU.

- If the data array is not empty, and the number of samples in the array is sufficient, the manufacturer server checks the number of samples for the same

brand and model. If the number of samples for the same brand and model is insufficient, the manufacturer's server completes the remaining number of samples required for adequate data by adding data from a different brand but the same segment. The manufacturer's server then responds to the OBU with the "valid data" information and sends the dataset containing the examples to the OBU.

3.1.4 Authentication

When the data request phase is completed, the OBU starts collecting sensor data from the vehicle's ECU to figure out the driver's driving behaviour. If there is relevant historical driving data received from the manufacturer's server for comparison, this collected sensor data from the ECU is analyzed using machine learning algorithms for identity verification. If there is no data available for comparison, these collected sensor data are only used for sampling the driver's driving behavior.

In the authentication phase of our proposed structure, we are not using raw sensor data to train a machine learning algorithm that creates a model for identifying the driver. Instead, as a first stage, the raw sensor data is processed by a Correlation-based feature extraction algorithm, resulting in a set of features being gathered.In the second stage, we train a machine learning algorithm with historical feature sets of driving. With this model, we query whether the current feature set belongs to the driver or not. The primary reason for implementing a two-stage analyzing method in our approach is to conserve memory by not storing raw sensor data on the blockchain and to create a shared dataset with the same data format for all vehicles. The flowchart of the two-stage analyzing is shown in Figure 10 to illustrate the working principle. Additionally, details about the stages are explained below.

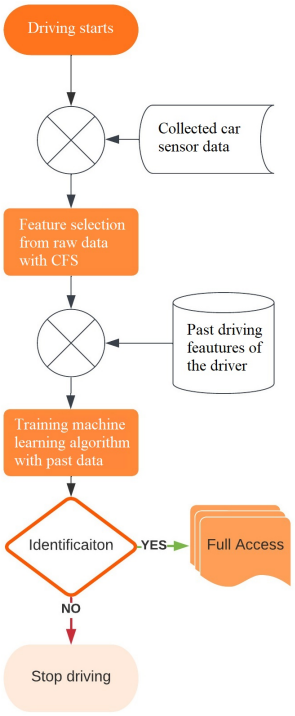

Figure 10: Authentication Flow Chart

3.1.4.1 Feature Selection

At this stage, feature selection is performed on the data collected from the vehicle for driving behaviour analysing. With this feature selection, a new data with lower dimensionality is obtained from the raw data, and redundancy in the data is minimized during the application of this method. In this stage, a CFS (Correlation-based Feature Selection) algorithm is applied to create feature set. The CFS algorithm is a technique used to select sub-features of the data, based on their correlation with the target variable. It also minimize the redundancy in selected variable.

CFS has basically three steps for processing the data. In the first step, the algorithm calculates the correlation coefficient values of each feature with the target value. Features that have higher correlation coefficient values are assumed to be more relevant to the target.In the second step, the algorithm considers the inter-correlation among the features themselves to avoid redundant features. In the last step, with the determined threshold value, CFS selects a balanced relevance with the target value

by avoiding redundant features.

3.1.4.2 Driver Identification

The second stage of our authentication structure, called as the decision stage, runs a machine learning algorithm. At this stage, the machine learning algorithm is trained with the feature dataset of historical drives that sent from the blockchain. After the model is trained, the feature set obtained from the current driving data is tested in the model to evaluate whether it belongs to the driver. If the feature set belongs to the driver, the authentication is considered successful. Machine learning algorithms perform different performance depending on the dataset. It is not expected that a machine learning algorithm will perform optimally on every dataset. Therefore, it is necessary to determine the algorithm that performs best with used dataset. During performance evaluation, machine learning algorithms previously used in relevant studies are trained with our dataset, and the optimal algorithm that performs best with our dataset is chosen. Multi Linear Regression, Decision Trees, and K-Nearest Neighbors algorithms were compared, and it was observed that optimal performance was achieved with the K-Nearest Neighbors algorithm. Therefore, MALBUAV authentication structure used the kNN algorithm as the machine learning algorithm for identity detection.

3.1.5 Blockchain Data Record

The model parameters obtained after each drive are recorded on the blockchain for the purpose of sampling the driver's driving behavior. For a data to be recorded on the blockchain, the driver's identity verification process must be successful or the OBU must have collected the data for sampling purposes. Driving features that have been compared with past driving features data and have failed the identity verification are not recorded on the blockchain. The data recording process after the drive is illustrated in Figure 11.

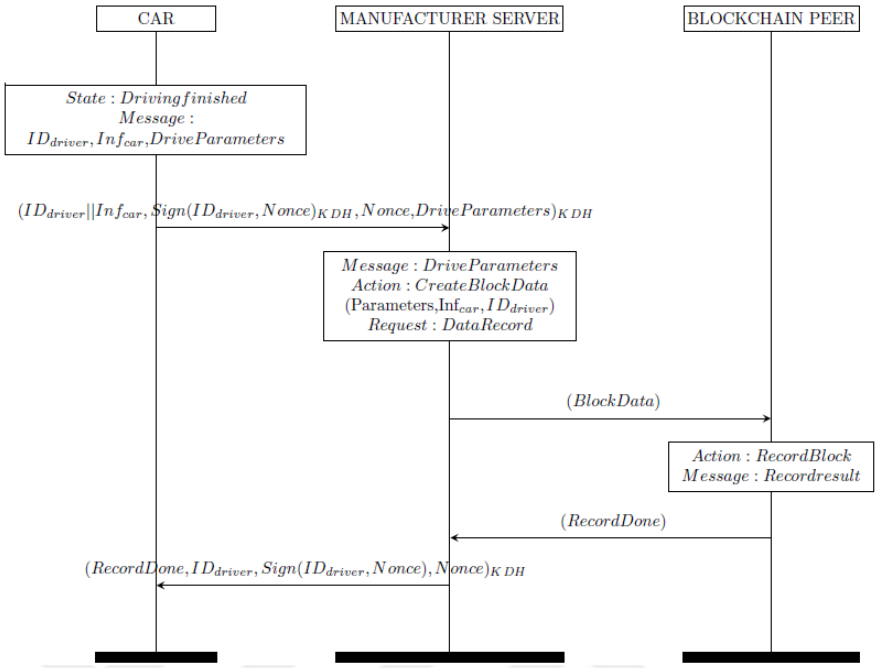

Figure 11: Data Record Squence

At the end of the drive, the OBU encrypts the vehicle's information, the driver's license IDs, and the driving features of the driving behavior model, and transmits them to the manufacturer's server. Due to security issues, the message also contains the signature and nonce values. The AVISPA result of this protocol is shown in Figure 12

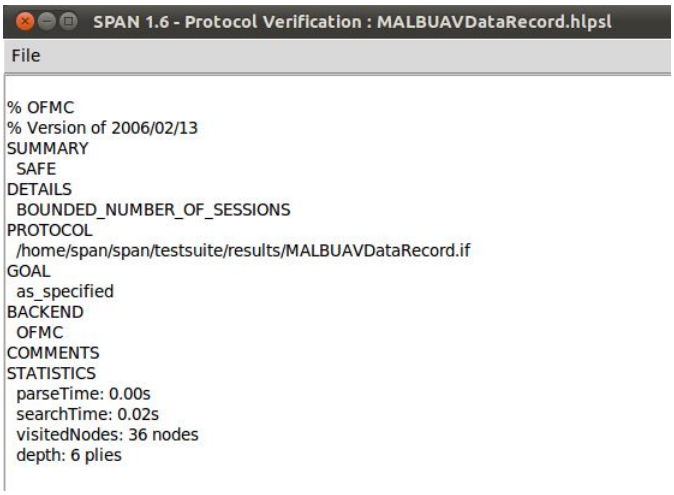

Figure 12: AVISPA Tool Data Record Results

The manufacturer server checks the security credentials and creates blockchain data that includes car info, driver info, and the feature set gathered from the raw data of driving for saving to the blockchain. After creating blockchain data, the manufacturer server sends it to the peer of the Hyperledger blockchain belonging to the manufacturer. The peer calculates a nonce for the block and creates the block, then broadcasts it to the orderer to save it in the Hyperledger blockchain. After saving is completed, the peer sends a "done" message to the manufacturer server. Then the manufacturer server informs the car that saving is complete.

CHAPTER IV

DATA AND TEST ENVIRONMENT

We created a simulation environment to test the proposed authentication system. In this test environment, we developed different software programs to simulate the components of the system, including the blockchain, manufacturer server, RSU, and the vehicle (OBU and ECU). We integrated various programmable devices into the test environment. Below, we explain the test environment prepared for each component separately.

4.1 OBU & ECU

For simulating the car, two Raspberry Pi programmable electronic devices and one Raspberry Pi touchscreen were used to model the car. These devices act as the OBU and ECU of the car. The two Raspberry Pi devices connect to each other over serial communication, and the device acting as the OBU sends commands to the other to inform the authentication result. In Figure 13, multimedia screen of the simulation environment is shown which created by Pi devices and touch screen.

Figure 13: OBU&ECU Simulation

Due to the hardware specifications of the Pi device, which have similar capacity to real OBU devices, we had a chance to observe the real performance of the proposed structure. The Pi devices run machine learning algorithms and the CFS algorithm to analyze the driver's driving behavior. Additionally, it communicates with a real computer over Wi-Fi, which simulates the manufacturer server.

4.2 VANET Modelling

In the VANET level of the framework we propose, both stationary and mobile components are present. Stationary components include manufacturer servers and RSUs, while the mobile components consist of vehicles. In Figure 14, we depicted the structure we created for the VANET simulation of our proposed study.

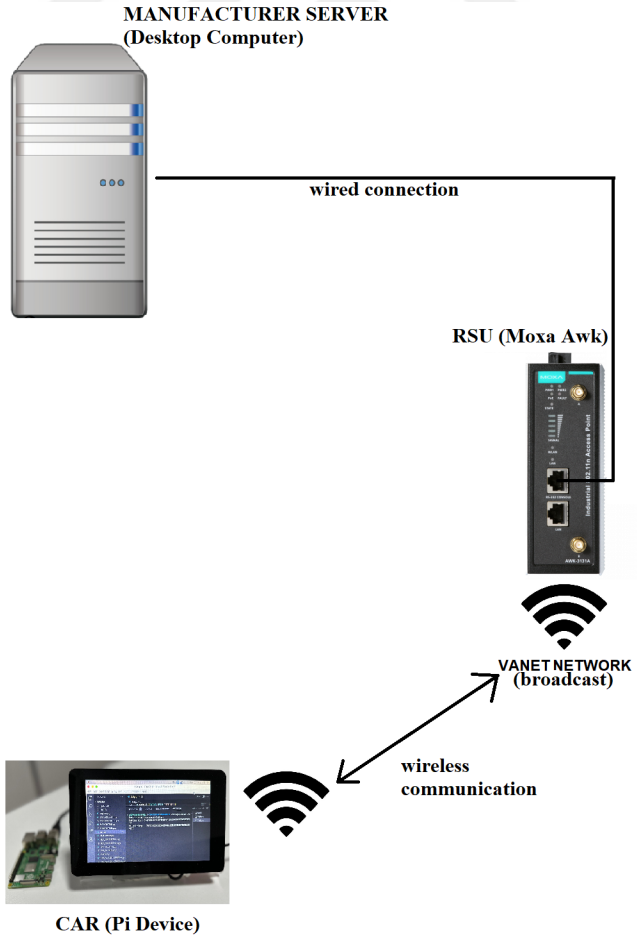

Figure 14: MALBUAV VANET Environment Simulation

In the previous section, we explained that we use PI device to represent the OBU of the car. In VANET level simulation in Figure 14, the PI device act like car and connects to MOXA AWK devices with wireless communication. MOXA device is an access point which represents the RSU of VANET level. MOXA broadcasts a wireless network, named as VANETNetwork", to provide wireless connection point for cars. RSU and server are stationary units and connects each other with wired connection. In our simulation, MOXA devices and a desktop computer which represent the manufacturer server communicate with wired connection. With this setup, we create a VANET network to test our proposed structure. The setup enable a communication for both stationary and mobile components.

4.3 Blockchain Modelling

A blockchain is a network where multiple devices act as peers. Since simulating this environment with real devices would be costly, virtual machines have been set up and programmed to act as peers. We used a service provider that provide virtual machines instead of installing virtual machines on a physical device. Using a service provider enabled quick setup both in terms of device provisioning and automatic package installation during setup. Virtual machines installed on Amazon Web Services have been set up as node and master devices, and the blockchain environment has been established on these machines.

In the blockchain simulation, different blockchain platforms have been examined. Among them, HYPERLEDGER FABRIC blockchain structure has been chosen due to its modularity, permissioned network, high performance, and scalability. With this structure, only manufacturer servers are included in the private permissioned network. The system allows manufacturer servers to send queries to the blockchain peers to access the driver's historical driving data. The scheme of the HYPERLEDGER FABRIC is shown in Figure 15 which provide explanation about relation

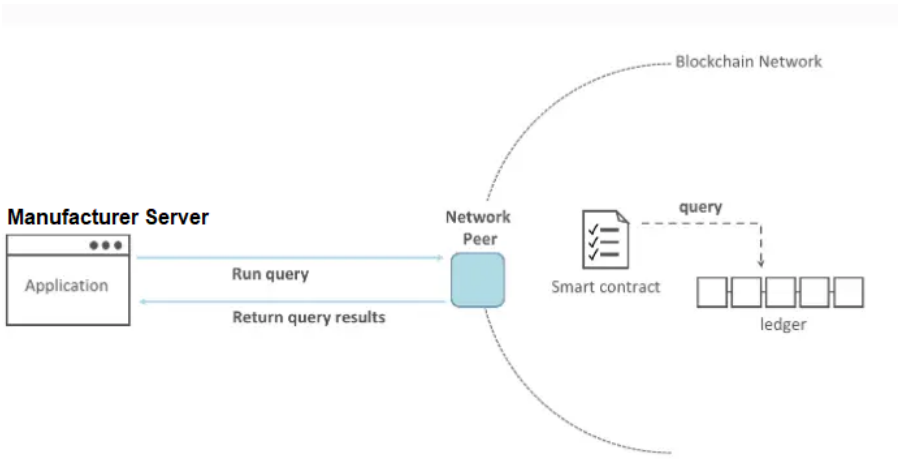

Figure 15: HyperLedger Fabric Network Scheme

between application and a peer.The application runs on the manufacturer server and communicates with peer that belong to the manufacturer server. It sends queries to them to get or save data.

4.4 Test Data

Our proposed authentication structure aims to securely authenticate in an ecosystem where different vehicle manufacturers, different segments of vehicles should exist and every driver can use any vehicle. Therefore, we needed a test dataset that includes the same segment but different brands vehicles, different segment but the same brand vehicles, and drivers who have used various vehicles within this group. The driving behavior dataset in [30] almost meets these mentioned variations. In this dataset, data were collected with three different vehicles (1.4 CC Ford F, 1.2 CC Ford F, and 1.2 CC Hyundai I20) and three different drivers. This dataset includes different brand vehicles from the same segment, and different drivers, making it nearly suitable for testing our system. Additionally, since data were collected using the MPU6050 sensor in all vehicles, there is no need for extra optimization for the data collected based on driving behavior. As the sensor is the same, the data will consist of relatively comparable values for driving behavior.

CHAPTER V

PERFORMANCE RESULTS

We focused on observing whether the recommended authentication structure provided the targeted improvements during the performance test. Therefore, we selected our performance metrics to observe the improvements made by the MALBUAV authentication framework in the system's hardware, the performance of algorithms, and the database segment.

In this evaluation process, our simulation environment and dataset were utilized to test our proposed system. While we do not have a large setup which will make stress test, it will enable us to evaluate the targeted improvements through the performance metrics we have selected. We can classify the selected metrics for the performance analysis of our system into the effects on vehicle hardware, the effects on authentication results, and the effects on the blockchain. When examining the effects on vehicle hardware, metrics such as the impact of raw data and CFS features usage in OBU's memory, changes in encryption times, and changes in the execution time of the algorithm were investigated. These changes were analyzed by reading the device's memory and processing time directly from the hardware. When examining the effects on authentication algorithms, the impact of using raw data and CFS features on the runtime of the algorithms, as well as their effects on accuracy, precision, and F1 score results, were investigated. These values were obtained by extracting algorithm performance report outputs after training the algorithms. When examining the effects on the blockchain, the accumulation degree of CFS features and raw data, as well as the Transactions Per Second (TPS) values of the blockchain, were investigated. To observe data accumulation, we generated thirty random driving data sets

and recorded these data sets as raw and CFS data. This thirty random driving set created by selecting random rows from test data that we used to run our simulation. We simulated the data accumulation that would occur by sequentially recording the generated raw data and CFS data for the randomly generated drives. To measure the TPS of the blockchain, batch size and batch time duration were sequentially changed to observe the maximum TPS output.

In our simulation, when the drive starts, the ECU sends seven different sensor data to the OBU every second. The ECU reads the dataset, which is used for testing in our simulation, from a file and sends the data of the chosen driver to the OBU. These sensor data are stored in memory until the analysis begins and processed during the analysis. If it is necessary to transmit the data to the database after the analysis, the data continues to be kept in memory. Storing this data on devices with low memory and processor hardware like OBU leads to performance degradation. In the proposed structure, we observed the impact on hardware of the OBU of training the machine learning algorithm using CFS data during the authentication phase and directly training machine learning algorithms with raw data. We observed the change in OBU memory over a period of twenty minutes from the start to end of the drive. This time interval consists memory usage of driving data in collection phase, in analysing phase and recording phase. The memory evolution which starts with driving and continue until driving finished, represented in Figure 16. This memory evolution consider driving data and past driving data of driver, the algorithm memory usage is not included. In the graph, the initial difference between raw and CFS data indicates the need for extra memory to store ten past driving data, if you use raw data rather than CFS data. Until minute 10, the OBU collects sensor data from ECU of the car for each method and the increase in memory is same. Since the size of the collected data is the same in both scenarios, it is expected to have the same increase. After minute 10, our work apply CFS to collected data and gets new feature set and

then deletes the collected data. Because of that in minute 11 the memory usage decrease in our work while it stay same in raw data usage. The reason for the memory usage remaining the same in raw data usage is the necessity of using raw data for testing, as machine learning algorithms are trained with the driver's historical raw data in this situation. In analysing and driving section which refers minutes between 11 and 17 the memory usage stay same for each situation.Because, both raw data and CFS data used for analysing the driver behaviour. In minute 17, analysing phase finishes and then past driving data that received at the start of driving is deleted in both situation. Because there is no need to store them in memory after driver identity verification is done. After 18 minutes, the remaining memory usage refers to the data size that will stay in memory to be sent to the server for the last driving behaviour data record. In Figure 16, the area under the lines of CFS and raw data represents memory usage for each. As seen in this graph, applying CFS to raw data has resulted in reduced memory usage throughout the duration from the beginning to the end of the drive.

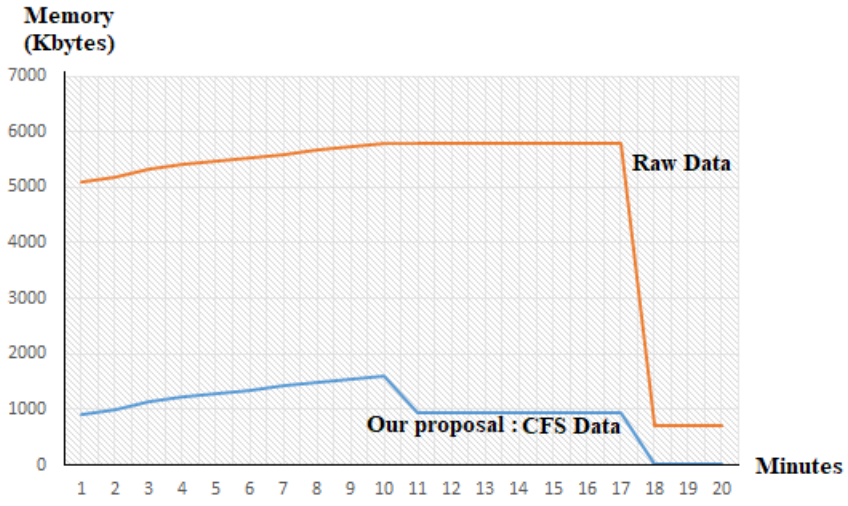

Figure 16: Memory Evolution of Dataset in OBU

In the process of identity verification, the OBU (On-Board Unit) communicates with the vehicle manufacturer's server to access the driver's historical data. To ensure

the security of this communication, the manufacturer server initiates a key exchange protocol with the OBU. In this protocol, they generate an asymmetric key pair for Diffie-Hellman key exchange, and at the end of the protocol, they securely share symmetric keys. At the end of the drive, the OBU encrypts the collected data using a symmetric key and sends encrypted data to the server. Throughout this process, both incoming and outgoing data undergo symmetric encryption and decryption. The encryption time for both incoming and outgoing data varies depending on the size of the data.

Table 1: Encryption Time of Driving Behaviour Data

Encrypted Data	**Time (sec)**
Raw Data 10 minute	0.0029
Raw Data 20 minute	0.0039
Raw Data 30 minute	0.0099
CFS set 10 minute	0.0009
CFS set 20 minute	0.0009
CFS set 30 minute	0.0009

Table 1 shows the encryption time of raw data collected after ten, twenty, and thirty minutes of driving and also the CFS features data encryption time of these drivings. In this table, as the driving time increases and thus the data size increases, the encryption time also increases. In proposed structure CFS selects features from the entire duration of the drive, so the data size remains unchanged. Therefore, the encryption time is lower, predictable, and almost constant compared to other scenarios.

During the generation of asymmetric keys using the RSA method, memory usage and time for production of key depend on the size of the prime numbers generated. As the size of the prime numbers decreases, memory usage decreases and also key production get fast, but this compromises the reliability of the encryption. In RSA,

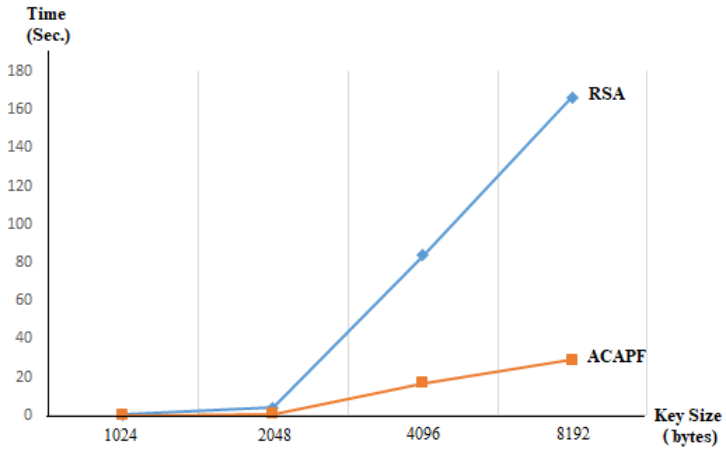

Figure 17: RSA and ACAPF Key Generation Times

finding prime numbers for calculating keys larger than 2048 bits could take minutes. Therefore, a modified RSA method called ACAPF is used for key generation in asymmetric key production. ACAPF uses four small prime numbers instead of a single large prime number like RSA and requires less hardware power and time compared to RSA. Figure 17 illustrates the time taken during the use of RSA and ACAPF to generate an asymmetric key, showing that ACAPF requires less time with same hardware for generating a key of the same size. In this simulation we run python and used a laptop which has intel i7 processor and 16GB RAM.

Table 2: Performance Table of Four Different Algorithms

Applied Algorithm	**Accuracy**	**F1 Score**	**Recall**
K-Nearest Neighbor	0.93	0.9	0.91
Decision Tree	0.87	0.86	0.865
Multi Lineer Regression	0.758	0.84	0.82
K-Nearest Neighbours +CFS	0.99	0.9	0.98
Decision Tree +CFS	0.93	0.9	0.91
Multi Linear Regression +CFS	0.70	0.76	0.74

In driver authentication, one of the most critical factors is the reliability of the algorithm's that determine the identity of the driver. The K-Nearest Neighbors,

Multi Linear Regression, and Decision Tree algorithms listed in Table 2 were trained with both raw data and data obtained with CFS. The performance of the models is indicated in the table. The accuracy value, which indicates the percentage of correctly predicted drivers from the test data by the trained models, was highest in the K-Nearest Neighbors algorithm trained with raw data. It was observed that training the kNN algorithm with CFS data instead of raw data led to an increase in accuracy prediction. F1 score is one of the most important metrics in determining the reliability of an algorithm. This metric answers how many true labeled examples are correctly identified as positive. It is derived from the division of true positives, the correctly predicted positives, by the sum of true positives and false negatives, the falsely predicted negatives. This value is observed in cases where missing a positive truth is more crucial than misclassified a negative example. In our case, if we cannot classify the correct driver data with the correct driver class, the driver may not be able to authenticate with their driving behavior. Therefore, it is an important metric for us. Additionally, the K-Nearest Neighbor algorithm yields the best F1 score in the results. Moreover, in the results of the kNN algorithm trained with CFS data, we could not observe a change in the F1 score with decimal sensitivity. Recall metric is also an important measure in the performance of the algorithm. It indicates the probability of capturing positive examples during prediction. If the recall value gets higher, the probability of correctly matching the driving behavior with the correct driver also increases. The results in Table 2, it is observed that once again the kNN algorithm has the highest recall value both with raw data and with CFS data. Additionally, it is observed that the recall value of kNN trained with CFS is higher than that trained with raw data.

Algorithm's processing time is also an important metric. An algorithm that runs faster on the same hardware will require less processing power. Since OBUs are low-processing devices used in vehicles, a faster algorithm is always preferred. Table 3

Table 3: Hardware Load Comparison Table of Four Different Algorithms

Applied Algorithm	**Memory Usage (MB)**	**Execute Time (sec.)**
K-Nearest Neighbor	92	7.26
Decision Tree	160	5.2
Multi Lineer Regression	56	5.4
K-Nearest Neighbours+CFS	88	7.06

provides the execution times of the algorithms run for analysis. In this table, Multi Lineer Regression and Decision Tree have again provided faster results with raw data compared to K-Nearest Neighbours, but the reliability of these algorithm is not higher as seen in Table 2. The K-Nearest Neighbours and Correlation-based Future Selection algorithms together runs faster than the K-Nearest Neighbours algorithm which trained with raw data. As a conclusion, CFS improves the performance of classification methods such as kNN and Decision Tree algorithms, while it degrades the performance of Multi Linear Regression, which is a regression algorithm. This situation may arise from converting a multi-class dataset into a binary-class dataset to apply the Multi Linear Regression algorithm for classification. In this process, incorrect driver classes are assigned zero, while the correct driver class is assigned a one notation. Due to feature selection, where the dominant features of the wrong drivers form the zero class, the regression method applied to learn this dataset tends to underfitting with the available data. Additionally, classification methods have achieved better results with dominant features.

We mentioned that the raw data is the sensor data collected during the drive. The size of this data varies depending on the driving duration and the number of sensors. Although the number of sensors is known, not knowing the driving duration results in uncertainty about the size of the data that will be obtained at the end of the drive. The unpredictability of the data size leads to uncertainty regarding the capacity needed in the database to store the raw data as driving data. Therefore,

systems using raw data are vulnerable in terms of database management.

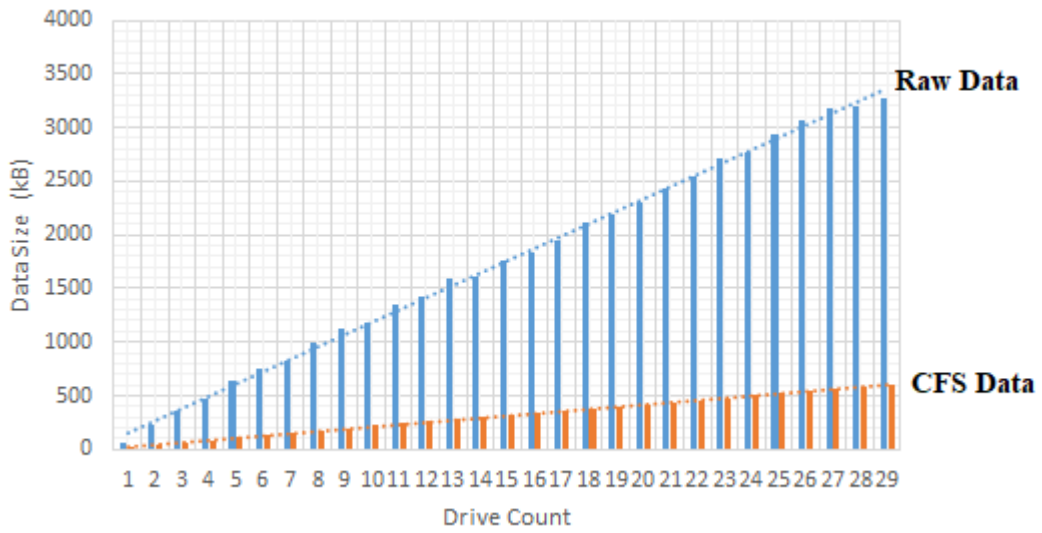

Figure 18: Data Accumulation in Blockchain

In the studies we examined, we noticed that they did not test the accumulation caused by the use of raw data in the data storage part. Therefore, in our study, we simulated the accumulation by scheduling thirty different drives with random usage duration in the database, and it is shown in Figure 18. This graph clearly shows that the increase trend in raw data usage is much sharper compared to the data increase trend in the CFS data usage. Therefore, proposed structure enables more robust estimation in database capacity management.

We used Hyperledger Fabric as the database, which is a blockchain, in our proposed architecture. For the simulation of this database, we set up Ubuntu machines on Amazon Cloud Service to simulate the peers. The manufacturer servers communicates to perform data querying and data recording processes with a peer. The performance of a blockchain is measured by the number of transactions it can process per second. This metric, called Transactions Per Second (TPS), varies depending on the batch time and batch size values of the blockchain. Batch size can be defined as the number of transactions grouped in a single block. Batch time, on the other

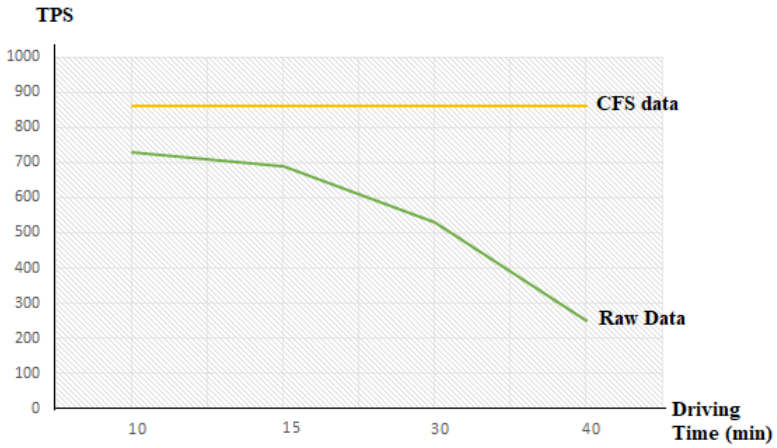

Figure 19: TPS evolution with Raw and CFS data

hand, is defined as the time it takes to create a block. If we clarify these variables with an example, we can assume that a transaction is like a cake batter and a block is like an oven. The number of cake batters that the oven can handle at one time is the batch size, and the cooking time for cake batters inside the oven is the batch time. So, the parameters of Hyperledger Fabric which are batch time and batch size need to be adjusted, and the transaction capacity it can handle per second needs to be known in order for the database to be able to respond to requests in a real VANET environment. In our case, the preferred batch size for Hyperledger Fabric is 524 Kbytes, and the raw data for ten minutes of driving data is 64 Kbytes. This means that a block could be created with eight sets of driving data if we save raw data as driving data for a 10-minute drive. If we increase the driving time to 15, 20, or 40 minutes, the raw data size will increase, resulting in batch sizes consisting of 6, 4, or 2 sets of driving data respectively. In the system we propose, the CFS data size is a maximum of 23 Kbytes for each drive. This size is independent of the driving duration. If we save CFS data as driving data, we can reach a maximum of 22 driving data in a block. Because CFS data is independent of driving time, we could create a block with 22 driving data even if the driving time increases to 15, 20, or

40 minutes. In Figure 19, we keep the batch time as 5 seconds and change the batch size to 8, 6, 4, 2 for raw data usage and 22 for CFS data. This batch size is related to 10, 15, 20, and 40-minute drives. In this simulation, the TPS value decreases for raw data usage, while it stays stable for CFS data usage. As a result, using CFS data would provide higher TPS value under the same batch time intervals. Additionally, this TPS value would be more predictable because the data size is independent of the driving duration. But in raw data, the number of transactions in a block varies according to the driving duration, making the TPS value unpredictable.

CHAPTER VI

CONCLUSIONS

In our work, we have introduced a framework where all vehicle manufacturers, different, can share the same environment and allow drivers to authenticate themselves while using vehicles from different brands. In this framework, trust has been established for different vehicle manufacturers to exchange information with each other with blockchain. Integrating raw data as driving data into blockchain indeed creates a high need for data capacity. Moreover, the variability in driving duration directly affects the amount of data to be recorded, introducing uncertainty in the required capacity. Therefore, in our study, we applied Correlation-based Feature Selection to the raw driving data, and these features were recorded in the blockchain as driving data. The effects of this method on the blockchain were examined in the simulation, revealing that it increased the Transaction per Second (TPS) value of the blockchain and added predictability in terms of both capacity needs and TPS value. The impact of recording values obtained from CFS as driving data on authentication algorithms has also been examined. During the authentication process, models trained with CFS data were compared with models trained with raw data, resulting in increased accuracy, F1 score, and recall values for the models. In the simulation, we achieved %99 accuracy using the kNN algorithm. At the end of the study, we have established an environment where all manufacturing companies can share information with confidence. This environment offers faster, more accurate, and more predictable results. Additionally, it enables each driver to access previous driving data for authentication purposes across different vehicle brands.

REFERENCES

[1] M. u Farooq, M. Waseem, S. Mazhar, A. Khairi, and T. Kamal, "A review on internet of things (iot)," *International Journal of Computer Applications*, vol. 113, no. 1, pp. 1–7, 2015.

[2] A. El Mekki, A. Bouhoute, and I. Berrada, "Improving driver identification for the next-generation of in-vehicle software systems," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 8, pp. 7406–7415, 2019.

[3] D. Zheng, C. Jing, R. Guo, S. Gao, and L. Wang, "A traceable blockchain-based access authentication system with privacy preservation in vanets," *IEEE Access*, vol. 7, pp. 117716–117726, 2019.

[4] R. Kaur, T. P. Singh, and V. Khajuria, "Security issues in vehicular ad-hoc network(vanet)," in *2018 2nd International Conference on Trends in Electronics and Informatics (ICOEI)*, pp. 884–889, 2018.

[5] Y. Xun, J. Liu, N. Kato, Y. Fang, and Y. Zhang, "Automobile driver fingerprinting: A new machine learning based authentication scheme," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 2, pp. 1417–1426, 2020.

[6] A. Burton, T. Parikh, S. Mascarenhas, J. Zhang, J. Voris, N. S. Artan, and W. Li, "Driver identification and authentication with active behavior modeling," in *2016 12th International Conference on Network and Service Management (CNSM)*, pp. 388–393, 2016.

[7] H. P. Dai Nguyen and R. Zoltán, "The current security challenges of vehicle communication in the future transportation system," in *2018 IEEE 16th International Symposium on Intelligent Systems and Informatics (SISY)*, pp. 000161–000166, 2018.

[8] R. Shrestha, R. Bajracharya, and S. Y. Nam, "Blockchain-based message dissemination in vanet," in *2018 IEEE 3rd International Conference on Computing, Communication and Security (ICCCS)*, pp. 161–166, 2018.

[9] W. Liang, Z. Li, H. Zhang, S. Wang, and R. Bie, "Vehicular ad hoc networks: architectures, research issues, methodologies, challenges, and trends," *International Journal of Distributed Sensor Networks*, vol. 11, no. 8, p. 745303, 2015.

[10] M. Altayeb and I. Mahgoub, "A survey of vehicular ad hoc networks routing protocols," *International Journal of Innovation and Applied Studies*, vol. 3, no. 3, pp. 829–846, 2013.

[11] A. Casteigts, A. Nayak, and I. Stojmenovic, "Communication protocols for vehicular ad hoc networks," *Wireless Communications and Mobile Computing*, vol. 11, no. 5, pp. 567–582, 2011.

[12] . S. D. R. Nema, S., "Performance analysis of ieee 802.11p based vehicular communication networks under mixed traffic conditions," in *2019 5th International Conference on Advanced Computing Communication Systems (ICACCS)*, pp. 1137–1141, 2019.

[13] S. Al-Sultan, M. M. Al-Doori, A. H. Al-Bayatti, and H. Zedan, "A comprehensive survey on vehicular ad hoc network," *Journal of network and computer applications*, vol. 37, pp. 380–392, 2014.

[14] M. J. . J. S. Kaur, G., "Ieee communications surveys tutorials," in *Collision avoidance using vehicular Ad Hoc networks: A review*, pp. 2080–2109, 2016.

[15] P. Bellavista, G. Cardone, A. Corradi, and L. Foschini, "Convergence of manet and wsn in iot urban scenarios," *IEEE Sensors Journal*, vol. 13, no. 10, pp. 3558–3567, 2013.

[16] P. H. L. Rettore, A. B. Campolina, A. Souza, G. Maia, L. A. Villas, and A. A. F. Loureiro, "Driver authentication in vanets based on intra-vehicular sensor data," in *2018 IEEE Symposium on Computers and Communications (ISCC)*, pp. 00078–00083, 2018.

[17] A. E. Mekki, A. Bouhoute, and I. Berrada, "Improving driver identification for the next-generation of in-vehicle software systems," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 8, pp. 7406–7415, 2019.

[18] T. Banerjee, A. Chowdhury, T. Chakravarty, and A. Ghose, "Driver authentication by quantifying driving style using gps only," in *2020 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, pp. 1–6, 2020.

[19] S. Gupta, A. Buriro, and B. Crispo, "Driverauth: Behavioral biometric-based driver authentication mechanism for on-demand ride and ridesharing infrastructure," *ICT Express*, vol. 5, no. 1, pp. 16–20, 2019.

[20] M. Rahbari and M. A. J. Jamali, "Efficient detection of sybil attack based on cryptography in vanet," *arXiv preprint arXiv:1112.2257*, 2011.

[21] H.-T. Wu and G.-J. Homg, "Vehicular cloud network and information security mechanisms," in *2016 International Conference on Advanced Materials for Science and Engineering (ICAMSE)*, pp. 196–199, 2016.

[22] S. RoselinMary, M. Maheshwari, and M. Thamaraiselvan, "Early detection of dos attacks in vanet using attacked packet detection algorithm (apda)," in *2013 international conference on information communication and embedded systems (ICICES)*, pp. 237–240, IEEE, 2013.

[23] M. Roy, S. Deb, J. Saha, S. Dhang, P. Chaudhury, and A. Mallik, "Acafp: asymmetric key based cryptographic algorithm using four prime numbers to secure message communication. a review on rsa algorithm," 2017.

[24] C. Lai and Y. Ding, "A secure blockchain-based group mobility management scheme in vanets," in *2019 IEEE/CIC International Conference on Communications in China (ICCC)*, pp. 340–345, IEEE, 2019.

[25] Y. Cheng, S. Xu, M. Zang, S. Jiang, and Y. Zhang, "Secure authentication scheme for vanet based on blockchain," in *2021 7th International Conference on Computer and Communications (ICCC)*, pp. 1526–1531, IEEE, 2021.

[26] M. Salimitari, M. Joneidi, and Y. P. Fallah, "Bats: A blockchain-based authentication and trust management system in vehicular networks," in *2021 IEEE International Conference on Blockchain (Blockchain)*, pp. 333–340, IEEE, 2021.

[27] D. Moussaoui, B. Kadri, M. Feham, and B. A. Bensaber, "A distributed blockchain based pki (bcpki) architecture to enhance privacy in vanet," in *2020 2nd international workshop on human-centric smart environments for health and well-being (IHSH)*, pp. 75–79, IEEE, 2021.

[28] C. Dai, X. Xiao, Y. Ding, L. Xiao, Y. Tang, and S. Zhou, "Learning based security for vanet with blockchain," in *2018 IEEE International Conference on Communication Systems (ICCS)*, pp. 210–215, IEEE, 2018.

[29] A. Armando, D. Basin, Y. Boichut, Y. Chevalier, L. Compagna, J. Cuéllar, P. H. Drielsma, P.-C. Héam, O. Kouchnarenko, J. Mantovani, *et al.*, "The avispa tool for the automated validation of internet security protocols and applications," in *Computer Aided Verification: 17th International Conference, CAV 2005, Edinburgh, Scotland, UK, July 6-10, 2005. Proceedings 17*, pp. 281–285, Springer, 2005.

[30] S. TIWARI, "Driving behavior dataset," 2021. https://www.kaggle.com/datasets/shashwatwork/driving-behavior-dataset [Accessed: 2024].

VITA

Huseyin TUNCEL is a M.Sc Student working with the supervision of Dr.Kubra Kalkan and is concurrently Automation Software Executive in VESTEL HOME APPLIANCE Inc. He graduated from the Department of Mechatronics Engineering at Bursa Technical University in 2017. He conducted his thesis on determining driving characteristics from the driver's eye and mouth movements in vehicles; and on fatigue detection and reporting driving safety based on these characteristics. After graduation, he continued his career in the field of route planning software for autonomous vehicles and management software for automated storage systems. Currently, he is developing projects on machine learning, data monitoring in industrial fields, and working on data collection software. Therefore, he is interested in topics such as machine learning, data processing, data security, and blockchain as a data storage structure.