A study on preserving privacy of Internet of Vehicles (IOV) data in cloud infrastructure using homomorphic encryption

MSc Research Project
Masters in Cybersecurity

Victoria Mfon Atauba
Student ID: x20122837

School of Computing
National College of Ireland

Supervisor: Dr Niall Heffernan

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Victoria Atauba Mfon		
Student ID:	x20122837		
Programme:	Cybersecurity	**Year:**	2022
Module:	MSc Research Project		
Lecturer:	Dr Niall Heffernan		
Submission Due Date:	15/08/2022		
Project Title:	A study on preserving privacy of Internet of Vehicles (IOV) data in cloud infrastructure using homomorphic encryption		
Word Count:	6796	**Page Count:**	20

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Victoria Atauba Mfon

Date: 15/08/2022

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

A STUDY ON PRESERVING PRIVACY OF INTERNET OF VEHICLE (IOV) DATA IN CLOUD INFRASTRUCTURE USING HOMOMORPHIC ENCRYPTION

ABSTRACT

IOV provides a paradigm that aid the development of heterogeneous and distributed vehicular systems, in which the processing and storage of large volumes of data has led to the adoption of cloud computing services. Cloud computing offers advanced technological services such as secure storage of data stored in the cloud and on-demand access to data and resources. The sophisticated structure of cloud-IoV environments creates a serious concern about data utility at the expense of privacy. While today's state-of-the-art encryption systems preserve users' privacy, they also make it impossible to do meaningful calculations on encrypted data. The potential of homomorphic encryption is that it allows encrypted data to be sent to the cloud, which can subsequently be computed without having to know the secret key. Therefore, this research addresses the privacy concerns associated with IOV data stored in cloud by proposing the use of Homomorphic Encryption. In addition, digital signature is employed to add additional security layer to ensure data confidentiality, integrity, and authentication.

Key word: Homomorphic encryption, cloud computing, Internet of Vehicles (IOV)

1 INTRODUCTION

1.1 OVERVIEW

In the recent decade, there has been a great development in mobile communications which enables the interchange of data between humans and anything, anywhere and anytime. Several new technologies have emerged to form smart vehicles which have evolved significantly. The adoption of this kind of development in mobile communications of vehicles has led to a much anticipated reality in the coming year (Derdour *et al.*, 2019). Internet of Vehicle (IOV) is a flexible network technology which establishes communication between diverse networks with the use of various models like: Vehicle-to-Vehicle (V2V), Vehicle-to-Roadside (V2R), Vehicle-to-Humans (V2H), Vehicle-to-Sensors (V2S) and Vehicle-to-Infrastructure (V2I). IOV is an application of Internet of Things (IOT) and improves upon Vehicular Ad-Hoc Networks (VANETs) by incorporating cellular networks like Long Term Evolution (LTE) to provide a communication that is expansive and reliable. VANETs are made up of vehicles which are connected in an ad-hoc method and communicate with one each through data sharing; while IOV on the other hand, spans a larger network which involves elements such as people, objects and diverse networks. IOV treats vehicles as smart entities with various sensors and computational capability for collecting and sharing data about its environments, other vehicles and road conditions. The IOV assimilates humans, vehicles and networks to make transportation system more robust, secure, and safe, as well as to provide smart cities with a range of services (Rasheed Lone, Kumar Verma and Pal Sharma, 2021).

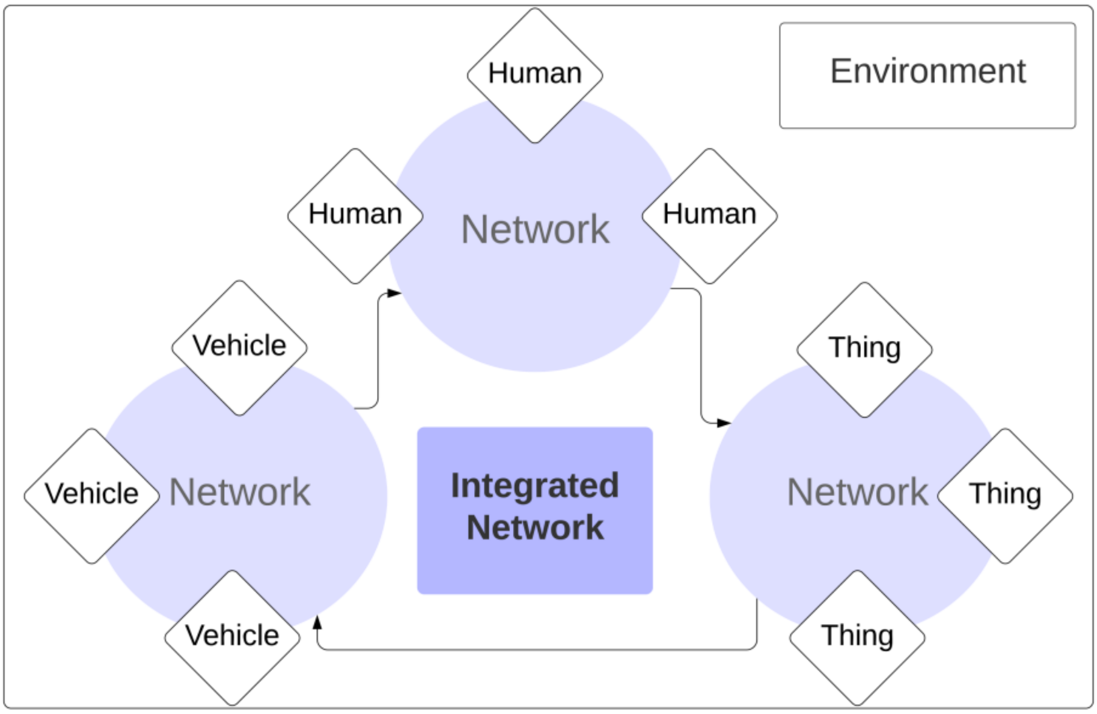


Fig 1: Network model of IOV

1.2 RESEARCH MOTIVATION

With the high demand of cloud computing, the cloud can be utilized to aid the abundant storage and computing of IOV data. The cloud can be adopted to help provide communication between vehicles by storing and processing data collected around smart cities. Cloud-based IOV does not only help in storing data, but offers useful services such as services from application providers and exchanging messages with other connected vehicles. However, the security and privacy concerns of cloud-based IOV, which encompasses confidentiality, data integrity and privacy preservation have been raised in recent times. As much as cloud service providers are very much honest and trusted, attackers can compromise privacy of IOV data by acquiring information like traffic situations from disseminated messages and broadcasting wrong messages to connected vehicles which can cause confusions and accidents. Disseminated messages should only be accessible by intended recipients, rather than all vehicles. If the security and privacy issues of cloud-IOV data is not handled properly, the development of cloud-IOV data may fall into a bottleneck (Huang *et al.*, 2019). It is certain that IoV will ultimately rank among the largest and most sophisticated networks when considering the enormous advantages it provides. In order to ensure the security and privacy-preserving of cloud-IOV data, homomorphic encryption is proposed in this study to guarantee the confidentiality of data and the computation and processing of data in encrypted form. This approach will help keep the privacy of connected vehicles and users intact, while offering maximum security and sufficient usability.

1.3 RESEARCH QUESTION

"How can privacy preservation of Internet of Vehicle (IOV) data be achieved using Homomorphic Encryption?"

1.4 RESEARCH OBJECTIVES

The major objective of this research is to extensively evaluate the effectiveness of adopting homomorphic encryption to secure IOV data, including the processing and storage of the data on the cloud. Therefore the structure of this report is as follows:

Section 1: Introduction to a brief overview of IOV, impacts of cyber attacks on connected vehicles, the motivation behind the adoption of cloud-based IOV and the objectives of the research.

Section 2: A detailed literature review of previously published academic journals and articles on privacy preservation, IOV, cloud security and homomorphic encryption.

Section 3-5: Discussion of the methodology, project design specification, and implementation.

Section 6: A presentation of the result evaluation, discussions and limitations.

Section 7: This section focuses on the conclusion and suggestions for further research.

2 RELATED WORKS

Numerous studies on cloud security flaws have been conducted since the development of cloud computing. IOV systems interact with their surroundings more often, which makes them more vulnerable to physical and digital threats. IOV domain security has been the subject of in-depth study that includes its enabling technology, architectural designs, deployments, and challenges.

2.1 CLOUD-IOV

Cloud computing provides a connection between objects and applications. (Mohiuddin and Almogren, 2020) highlighted that there have been persistent issues with the cloud service provider's lack of trust and the physical location of data which are transmitted by several IoV nodes to the cloud. (Amin *et al.*, 2018) explained that multi-tenancy storage architecture in cloud databases has also attracted criticism for storing a variety of users' data in one place, potentially endangering security and disclosing sensitive information. (Mohiuddin and Almogren, 2020) identified the insider threat as a type of threat linked to mistrust in the cloud service provider. The insider threat is one of the most unsolved issues with smart vehicles today. (Mohiuddin and Almogren, 2020) recognized a few significant challenges of cloud-IOV:

i Data Security and Privacy: Autonomous vehicle data is sent to the cloud for processing and retention. In the future, Cloud-IoV will be used extensively and permeate every aspect of a person's life. However, (Mohiuddin and Almogren, 2020) reveals that its widespread adoption is being hampered as individuals worry about the privacy of their data. In order to solve the difficulties of data privacy, (Henze *et al.*, 2016) developed a user-driven Privacy Enforcement for cloud-based services in the IoT. This is a user-oriented technique to ensure that the privacy of data produced by smart devices are preserved and processed in the Cloud.

ii Computational and Storage Performance: (Mohiuddin and Almogren, 2019) highlighted that while IoT devices are in motion for certain applications, processing and storing IOV data in the cloud requires a high level of performance criteria, which may be difficult to meet in all circumstances.

2.1 SECURITY AND PRIVACY CONCERNS IN IOV

IOV security concerns, including numerous attack types that comprise data availability, confidentiality, and privacy during information exchange between automobiles and users, have been studied by a number of researchers. (Han *et al.*, 2017) described the Sybil attack, in which malicious nodes create several false identities in order to steal information. The attacker nodes exploit the unique IDs that each node in the IOV possesses to gain access and steal data. The Resend attack is another sort of attack described by (Samara, Al-Salihy and Sures, 2010) in which the attacking nodes take the authentication information of legitimate nodes and resend them to the authentication node, therefore invalidating the authentication and achieving deception. Man In The Middle (MITM) attack discovered by (Zhou *et al.*, 2021) is another concern in which hostile control is imposed over channels of communication amongst two or more nodes and there is a tendency to intercept, tamper with, or replace the target's data. (Cheng *et al.*, 2021) stated that as IOV operates inside a wireless network environment, communications received by smart vehicles should be authenticated and safeguarded because it is simple for attackers to intercept, change, and transfer data. There have been a number of earlier attempts to address the security and privacy issues in IOV, however the bulk of these studies have been threat-centric and out-of-date rather than focused on the threats' defenses. For instance, the most recent study by (Contreras-Castillo, Zeadally and Guerrero-Ibanez, 2018) examined IOV's protocols, systems, and standards but did not give appropriate attention to the protective measures. According to (Huang *et al.*, 2019), some of the IOV's difficult issues include:

i MESSAGE ACCESS CONTROL IN IOV: A difficult issue of IOV for monitoring the access to dispersed messages is the message access control. According to (Xia *et al.*, 2017), using public key infrastructure (PKI) certificates to provide message access control in IOV is a good approach since it can guarantee vehicle-to-vehicle communication secrecy. However, in dynamic vehicle communication networks, it cannot ensure a perfectly suitable access control for multiple recipients. Additionally, (Xia *et al.*, 2017) explained that multimedia messages should be taken into account because information can be sent to surrounding automobiles for transmission. Studies have been done to estimate the computational latency compared to the PKI system. (Xia *et al.*, 2017) further suggested a CP-ABE delegation method that enables RSUs to handle the majority of calculations to increase the efficiency of the vehicles' decryption. Decision trees were also used to optimize a number of variables, including the distance from RSUs, communication costs, and computation costs.

ii MESSAGE VERIFICATION IN IOV: Another difficult aspect of IOV is message verification, which is accomplished via signature technologies like identity-based signature (IBS) and attribute-based signature (ABS). In order to overcome the key security vulnerabilities in IOV, (Li *et al.*, 2020) devised an approach that makes use of an effective message authentication with revocation transparency utilizing blockchain. He used a cuckoo filter, which increased efficiency by enabling RSUs to validate the signatures of neighboring vehicles. To guarantee the proposed approach satisfies all of the security and privacy standards of IOV networks, a security and efficiency assessment was also undertaken and its results were compared to those of current solutions. However, the work associated with producing keys and verifying certificates used more resources and processing power. By using attribute-based encryption to safeguard distributed messages and the deployment of a verifiable encryption and decryption method, (Huang *et al.*, 2019) in his study suggested a secure and privacy-preserving dissemination technique for disseminated messages in cloud-IOV in an effort to save computing time.

iii CONDITIONAL PRIVACY PRESERVATION: (Huang *et al.*, 2019) argued that drivers' identities and private information should not be shared, and listed strategies for storing drivers' data in tamper-proof containers or using pseudonym identities to secure vehicle identities. A batch verification for secure pseudonymous authentication (b-SPECS+) in IOV was proposed by (Horng *et al.*, 2013). With the use of an alternative method for creating signing keys, this technique aimed to fix the flaw in SPECS that was suggested by (Chim *et al.*, 2011). To meet the security and privacy requirements of smart cars, (Tzeng *et al.*, 2017) also put out the Identity-based Batch Verification (IBV) conditional privacy system.

2.2 APPROACHES TO IOV DATA PRIVACY PRESERVATION AND SECURITY

In order to ensure the protection of data privacy in IOV, certain researchers have put forward various solutions. (Butt *et al.*, 2019) examined the issues crucially needed to be considered for maintaining privacy in a Social IOV (SIOV) setting from several angles, such as the privacy of a person, behavior, communication, feelings, and location. Within a SIOV context, the authors of (Arora and Kumar Yadav, 2018) presented a blockchain solution that offers authentication and secure data transmission between cars and nodes. To ensure precise communication between nodes and things in the environment, a two-part technique was designed for the safe transmission of data. In the first approach, users will be required to register their vehicles with the Register Authority (RA), which then issues a Pseudo Identity (PIDi) and a public-private key pair. The secure data exchange was modeled using the second technique. To determine the PIDi of the cars, the message would first be encoded using the public key and then decoded using the private key. The PIDi is then authenticated if it is located on the blockchain. According to (Arora and Kumar Yadav, 2018), the main goal of using the blockchain system was to prevent unauthorized nodes from accessing messages. In addition, (Yahiatene *et al.*, 2019) have suggested a software-defined system that is also based on blockchain to protect networks and smart cars. Principal controllers, road-side units, and local controllers are the three different types of controllers that are required. The proposed Distributed Miners Connected Dominated Set (DMCDS) algorithm's performance was assessed using a number of variables, including mobility of nodes and trustworthiness. But the suggested method still appears to have problems, particularly in regard to security. In order to guarantee message integrity, non-repudiation, and information privacy, (Noh, Jeon and Cho, 2020) have also suggested a message authentication method within the blockchain. The suggested message authentication code, however, was insufficient to ensure the rigorous privacy of vehicle information. However, the recommended message authentication code for messages was insufficient to guarantee the privacy and rigid security of vehicle information.
s

In the IOV network, (Baldini *et al.*, 2019) have suggested a Blockchain-Based Zone Keys Trust management. In this study, the solution worked by confirming the good form of the blockchain data; if in good form, the public key infrastructure permits a certificate to be given to the vehicle alone. However, this method still had significant drawbacks, such as the difficulty of maintaining a tight level of security within the system and the dynamics of the network architecture. In order to design intelligent transportation systems, (Benomarat, Madini and Zouine, 2018) presented a cognitive-IOV paradigm method that blends cloud solutions and artificial intelligence together. (Benomarat, Madini and Zouine, 2018) went further to explain that the increased connectivity between multiple technologies is what makes IoV unique and that many processes still depend on human intellect. It has been determined that data analysis and knowledge discovery employing data mining, deep learning, and machine learning are essential to understanding the operation of cognitive engines. A number of problems with CIOV security and privacy were found by (Fida *et al.*, 2019), many of which are connected to the cloud. (Masood, Lakew and Cho, 2020) highlighted some of the attacks on networks which includes eavesdropping, data interception, data manipulation, man-in-the-middle attacks, flooding, and spamming.

2.3 HOMOMORPHIC ENCRYPTION

The conventional methods of encryption have failed in guaranteeing adequate security and is replaced by homomorphic encryption today. Shafi Goldwasser developed the concept of homomorphic encryption back in 1984. According to Shafi Goldwasser, data can be processed and worked upon without even revealing it or having to show it to another entity. The essence of the proposal of this concept was to deploy data encryption where it is required because, in the absence of an effective encryption mechanism, the data of individuals and businesses risk being exposed to significant security threats and loss. Goldwasser established homomorphic encryption technique to preserve data privacy and security of data. In this encryption, data is encrypted by the sender before being decrypted by the receiver, who then transmits the decrypted data back to the sender after applying the necessary mathematical operations. Homomorphic encryption has three different key types: public key, private key, and evaluation key. The private key aids in the process of decrypting data, the public key is used to encrypt data while the evaluation key aids in performing homomorphic operations. The private key serves as the primary building block for the evaluation key while the public key helps to control the increase of noise.

2.4 HOMOMORPHIC ENCRYRPTION TYPES

There are three types of homomorphic encryption, namely: Partially Homomorphic Encryption (PHE), Somewhat Homomorphic Encryption (SHE) and Fully Homomorphic Encryption (FHE) (Mohan, Kavitha Devi and Prakash, 2017). PHE is the kind of encryption that allows only specific mathematical functions to be performed on the given encrypted data, and consist of a single type of operation, either addition or multiplication. PHE does not limit to the depth or size of the circuit, allowing the user to freely interpret any encrypted data. Nevertheless, this type of encryption does not allow its users to use a wide variety of circuits and the only limitation to this encryption is that only a single kind of operation can be performed at one time. Somewhat Homomorphic Encryption (SHE) is the kind of scheme in which the encrypted data with operations available for both addition but limited level of multiplication. The problem with SHE is that unlike the PHE, this encryption has some limitations which prevent it from being excessively applicable, namely: the operations of any given encrypted data can only be performed a certain number of times, SHE limits the user to the depth of the circuit being not more than 5, and is mainly useful to evaluate low degree polynomials up to a certain criterion. Somewhat Homomorphic Encryption (Mohan, Kavitha Devi and Prakash, 2017). Fully Homomorphic Encryption (FHE) is similar to Somewhat Homomorphic Encryption (SHE). Fully Homomorphic Encryption can evaluate circuits with both addition and multiplication operations. But unlike Somewhat Homomorphic Encryption, this scheme can evaluate circuits with unlimited depths and this scheme is still in developmental stages today. Fully Homomorphic Encryption is the most widely used because it keeps information protected, accessible and is a powerful encryption scheme. The best feature of this encryption is that it protects your devices and data from being accessed or used by third-parties and protect devices from quantum attacks. Fully Homomorphic Encryption comes with the feature of protecting privacy without the need of the user to disable anything. The only problem with this scheme is that because it is not yet fully developed, it is slow in usage and has been observed quite frequently on heavy computers where the speed of the computer slows down (Mohan, Kavitha Devi and Prakash, 2017). In addition to that, two mathematical operations cannot be carried out in Fully Homomorphic Encryption because these two operations tend to cancel each other's effect, letting the efforts for security of data go down the drain. Cryptosystems like RSA are highly homomorphic in multiplication operations that are used all over the world today.

With homomorphic encryption, ciphertexts can be used for computation, generating an encrypted result that, when decoded, is identical to the outcome of the operations as though they had been carried out on the plaintext (Gaid and Salloum, 2021). (Matsumoto and Oguchi, 2021) described a technique that can use data safely and efficiently without applying pressure on smart devices. Somewhat homomorphic encryption (SHE), a lightweight advanced encryption standard, and Fully Homomorphic Encryption (FHE) were paired and used in the system. The findings of the experiment, according to (Matsumoto and Oguchi, 2021), indicated that the load on a smart device may be lowered to about 1/1400 when compared to the load of the system. (Ramesh and Govindarasu, 2020) devised a homomorphic encryption technique that encrypts IOV data using an unspecified number of computations. To provide security, a little amount of noise was introduced to each encrypted message, and it grew exponentially with each multiplication operation. (Ramesh and Govindarasu, 2020) addressed this by resetting the noise levels by bootstrapping the ciphertext. Bootstrapping, on the other hand, incurs significant performance costs, rendering the approach ineffective for real-world application. Furthermore, (Alaya, Laouamer and Msilini, 2020) explains that in the insecure cloud computing environment, homomorphic encryption has significant potential for privacy preservation.

Homomorphic encryption has proven to be very effective in the preservation of data stored in the cloud. In this study, Homomorphic encryption will be used to guarantee the security of data derived from smart vehicles and stored in the cloud. It will also be used to show the processing of encrypted data in its encrypted form without having to decrypt it in order to ensure the safety of sensitive data. The proposed solution will be examined extensively in the methodology section.

3 RESEARCH METHODOLOGY

This section highlights the detailed description of the proposed solution and methodology. Homomorphic encryption scheme is employed to guarantee data confidentiality and enables privacy preservation to be achieved, as it allows arbitrary computations to be performed on plaintext encapsulated inside a ciphertext without the potential threat of deanonymization. (Stefan and Slamanig, 2013) highlighted that enabling arbitrary computation on homomorphically encrypted data requires the design of a specific circuit representation for the algorithm at hand which may be a nontrivial task.

The functionality of homomorphic encryption can be described using two datasets of different classes in plain form. The first class contains a set of integers: 'x', 'y' and 'z', while the second class are logarithms of the integers. Homomorphic encryption can be applied by multiplying the integers and summing the logarithms of the integers. In this case, the multiplication of x and y = z, and (log) x + (log) y = (log) z. If 'x', 'y' and 'z' represent 'a', 'b' and 'c' in an encrypted form, and the same operations are carried out on 'a' and 'b' to give 'c', the answer 'c' will be equivalent to 'z'. Furthermore, homomorphic encryption comes with an inevitable problem which is referred to as noise. When a plain text is encrypted, a strong noise is followed and as operations are performed on the encrypted data, the noise multiplies further making it difficult to perform operations. According to (Gentry, 2008), if the encrypted data is cracked and is again encrypted, it can reduce noise to a large extent. However, the solution is not applicable as the cracking of encrypted data demands a secret key which is not easily available.

Therefore, in this report, to implement the homomorphic encryption solution, an algorithm is first designed and used to ensure the privacy preservation of IOV data stored on the cloud. To add an additional layer of security, digital signature is employed to guarantee data integrity, authentication, and non-repudiation. The data in which smart vehicles derive from its environments are first encrypted homomorphically at the field level and then digitally signed before dissemination. On the cloud, the vehicle which sends the information is first authenticated to verify that the information transmitted is from the right source and has not also been modified during transmission. Furthermore, necessary computations and processing are then performed on the information.

3.1 DATASET DESCRIPTION

The dataset utilized in this study which is primarily based on real data gathered by the General Department Council of Val de Marne (94) in France, is the vehicle mobility dataset. The dataset information includes a variety of elements, such as the road layout of Creteil Roundabout from the Open Street Map database, road traffic data derived from counting cars on the road and analysis from camera, traffic assignment of the vehicular flows, and traffic signal mechanism derived from manually recorded sequence adaptation in the regional transportation system. Included in the data are the vehicle slope, vehicle lane, vehicle angle, type of car, speed of the vehicle, and vehicle identification.

4 DESIGN SPECIFICATION

An experiment is designed to show how data confidentiality, integrity and authentication can be achieved using homomorphic encryption and digital signatures. For the simulation, the system designed consists of three actors: the client, the proxy-server, and the server. The client actor is responsible for the simulation of a real-life vehicle which sends the real-time data to the server. The server is where all the processing of the data in encrypted form is carried out. The server can further be deployed on the cloud, but for the sake of prototyping in this research, the server will be run in a local machine. The data being sent to the server for processing is routed via the proxy-server. The proxy-server is the point where there is a possible occurrence of network attacks, depending on the configuration of the simulation.

For the design of the circuit for homomorphic encryption, firstly, all the operations that need to be performed on the encrypted data will be identified. For example, an operation to determine if all the sensors present in a vehicle are sending valid readings. For each operation that is identified, an algorithm or set of algorithms will be devised to carry out each operation. And for each algorithm, a circuit with fully homomorphic encryption primitive will be developed to evaluate that algorithm with fully homomorphic algorithm (FHE).

For the simulation of the solution designed, the network functionality is intentionally baked into the proxy server to mimic an adversary. The main purpose of the proxy server is to relay all incoming messages from vehicle clients to the server, but it can also alter the messages probabilistically thereby causing a wide range of attacks. The proxy server has two modes: active and passive modes. In the passive mode, the proxy server simply eavesdrops on the communication while in the passive mode, it attempts to obstruct the communication or alter messages meant for the server. On the server side, all incoming messages are verified using digital signatures. If any transmitted payload or data modifications are detected at this point, the payload is simply dropped, otherwise, the server proceeds to process the payload.

5 IMPLEMENTATION

To implement the solution, a simulation environment was set up in Python programming language coupled with the use of the PyFhel open-source library for homomorphic encryption instead of reinventing the wheel. The simulation environment has a client-server setup that uses HTTP for communication. Python For Homomorphic Encryption Libraries, (Pyfhel) implements some functionalities of homomorphic encryption libraries such as addition, multiplication, exponentiation, and scalar products in python programming language. Pyfhel makes use of syntaxes similar to normal arithmetic such as: +, -, *. Pyfhel is built on top of Abstraction for Homomorphic Encryption Libraries (AHFEL) in C++ and serves as a common API for SEAL and PALISADE backends. Below is a diagram showing the simulation environment and different actors:

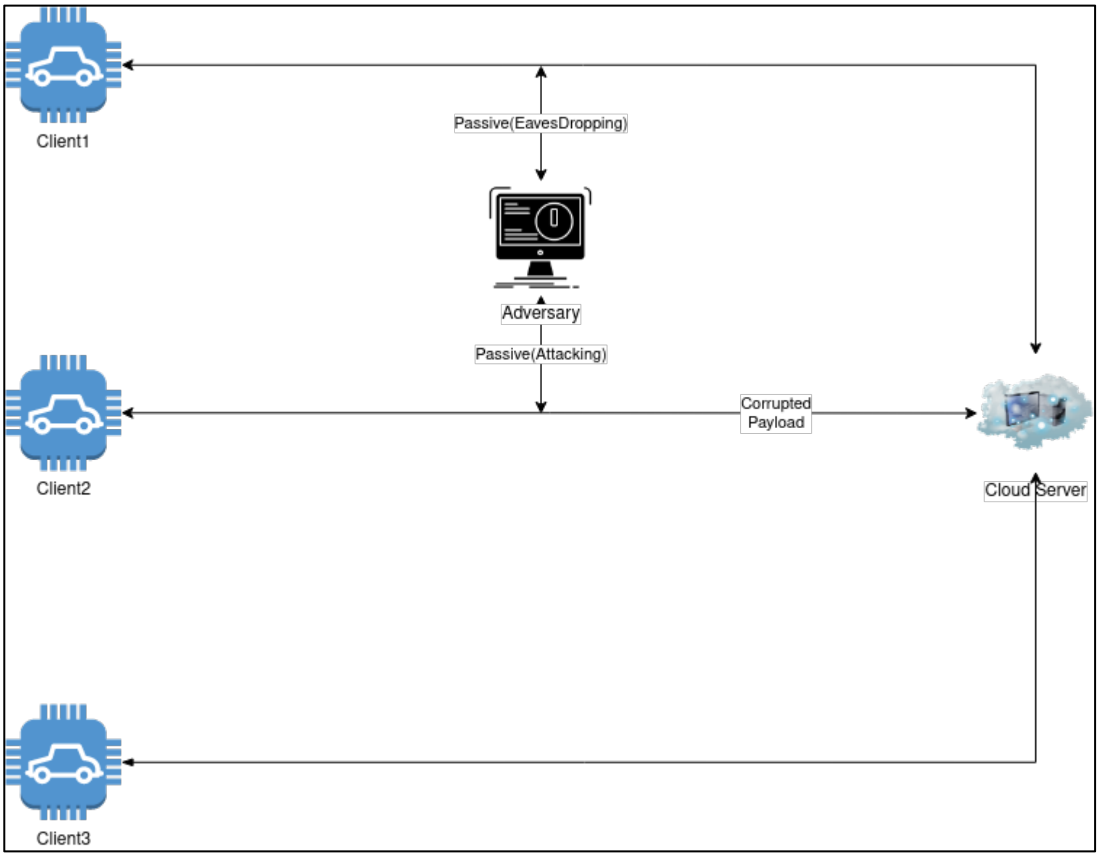
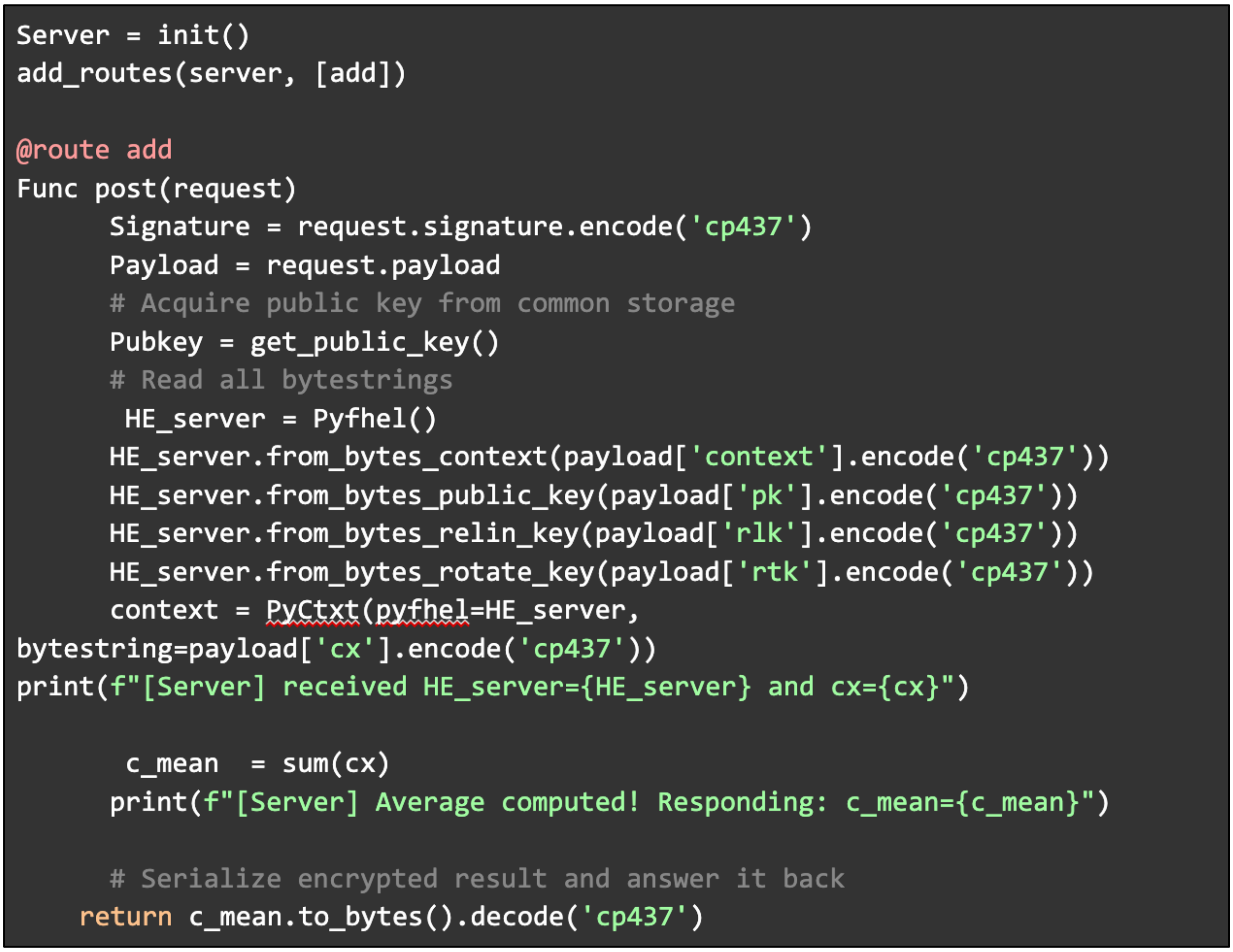


Fig 2: Architecture Diagram of the Simulation

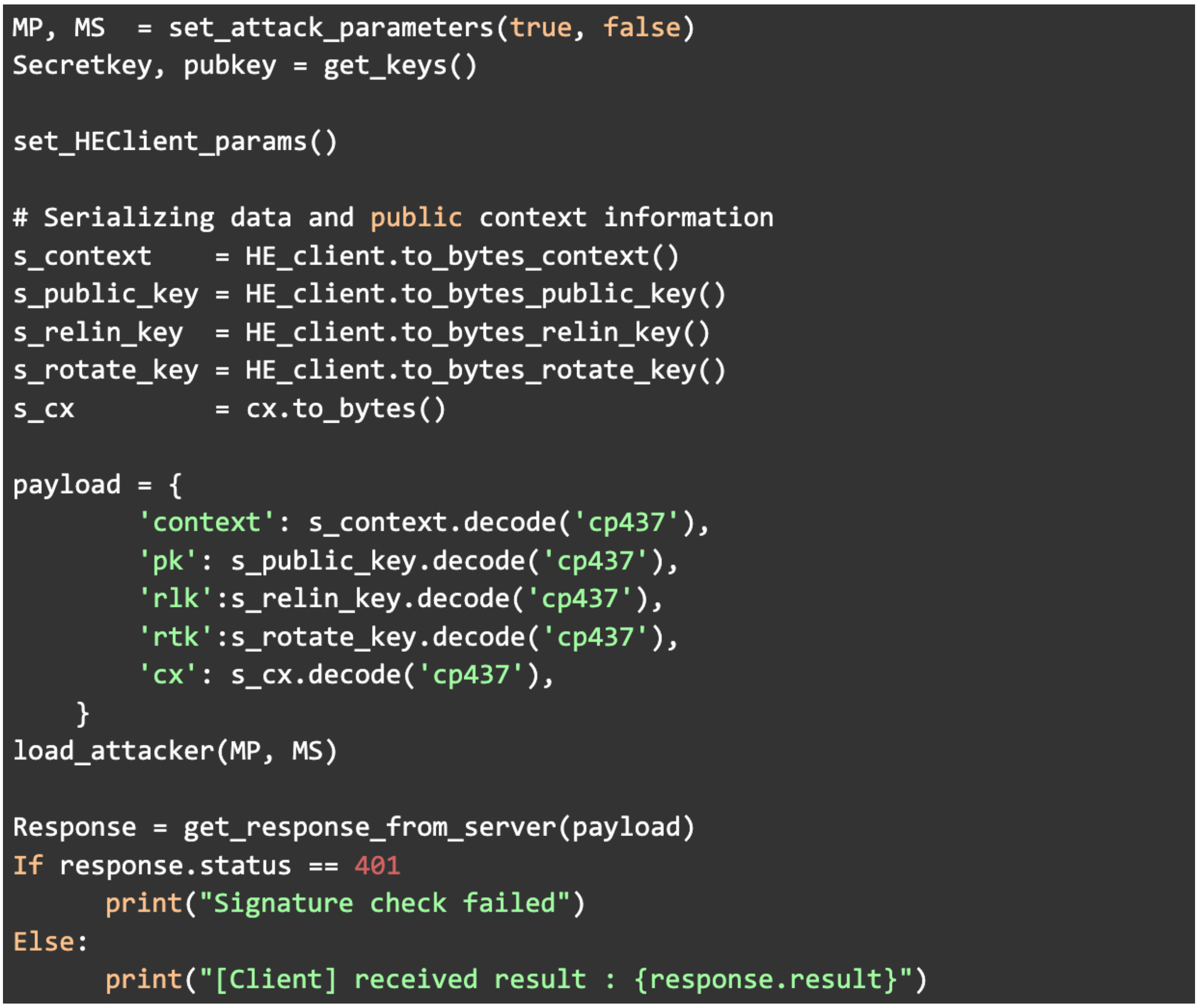
Upon initiation, the server starts listening for incoming connections on the local host. The server has different routes to support separate homomorphically encrypted operations. Whenever a request is received, first, it's digital signature is verified and the request is further processed only if the digital signature is valid.



```
Server = init()
add_routes(server, [add])

@route add
Func post(request)
      Signature = request.signature.encode('cp437')
      Payload = request.payload
      # Acquire public key from common storage
      Pubkey = get_public_key()
      # Read all bytestrings
       HE_server = Pyfhel()
      HE_server.from_bytes_context(payload['context'].encode('cp437'))
      HE_server.from_bytes_public_key(payload['pk'].encode('cp437'))
      HE_server.from_bytes_relin_key(payload['rlk'].encode('cp437'))
      HE_server.from_bytes_rotate_key(payload['rtk'].encode('cp437'))
      context = PyCtxt(pyfhel=HE_server,
bytestring=payload['cx'].encode('cp437'))
print(f"[Server] received HE_server={HE_server} and cx={cx}")

       c_mean  = sum(cx)
      print(f"[Server] Average computed! Responding: c_mean={c_mean}")

      # Serialize encrypted result and answer it back
    return c_mean.to_bytes().decode('cp437')
```

Fig 3: Pseudocode for the server working

On the client side, whenever the client requires some processing done on the server side, the client first prepares the payload by adding homomorphically encrypted data into it and then digitally signs the payload and includes the data in the payload.



```
MP, MS  = set_attack_parameters(true, false)
Secretkey, pubkey = get_keys()

set_HEClient_params()

# Serializing data and public context information
s_context    = HE_client.to_bytes_context()
s_public_key = HE_client.to_bytes_public_key()
s_relin_key  = HE_client.to_bytes_relin_key()
s_rotate_key = HE_client.to_bytes_rotate_key()
s_cx         = cx.to_bytes()

payload = {
        'context': s_context.decode('cp437'),
        'pk': s_public_key.decode('cp437'),
        'rlk':s_relin_key.decode('cp437'),
        'rtk':s_rotate_key.decode('cp437'),
        'cx': s_cx.decode('cp437'),
    }
load_attacker(MP, MS)

Response = get_response_from_server(payload)
If response.status == 401
      print("Signature check failed")
Else:
      print("[Client] received result : {response.result}")
```

Fig 4: Pseudocode for Client working

Furthermore, digital signature was then implemented to add additional security layer. A digital signature is a mathematical technique used to validate the authenticity and integrity of a message, software, or digital document. The RSA Digital Signature Scheme was used for this implementation. With the RSA digital signature scheme, each client in the system generates its own public and private key and publishes its public key into the ecosystem. Every payload the client sends to the server is signed using the private key and consequently can be verified using the public key on the server side.

5.1 ADVERSARIAL ATTACKS

Adversarial attacks on client-server communication were simulated. The primary focus of the attacks is based on the integrity of the payload sent by the client actor to the server. To recap, the payload contains homomorphically encrypted data and the digital signature of the entire payload. The first type of attack carried out is to intercept the payload and alter the data encrypted homomorphically. This attack type is an attempt to send malicious data to the server and portray it as the client has originally sent it. Because the client is in possession of its private key, the adversary can not forge the digital signature. On the server side, before the processing of the incoming request proceeds, the digital signature is first verified and in this case, the signature becomes invalid as the contents of the packet have been modified. Hence, this alteration attempt is detected and thus not processed any further.

In the second type of attack, the adversary intercepts the packet, modifies its homomorphically encrypted payload and adds an updated digital signature using its own key. When the server receives the request, it validates the signature and finds the signature to be invalid as the public key associated with the sender was not used to sign the payload. Consequently, this adversarial attempt also fails.

6 EVALUATION

To carry out the evaluation, three models were used, namely:

a. No encryption
b. Advanced Encryption Standard (AES)
c. Fully Homomorphic Encryption (FHE)

In each model, the communication is set up as laid out in the foregoing sections. The attack types discussed above are expanded into four configurations as shown below;
Attack type A < {(MP_f, MS_f), (MP_t, MS_f), (MP_f, MS_t) and (MP_t, MS_t)}
Where MP = Payload Modification
MS = Signature modification and the
Subscript t = true
Subscript f = false

For each of the modals, the four cases are evaluated for security parameters as shown below:
Security parameter P < {integrity, confidentiality, authenticity, availability, non-repudiation}
Also, the models are given a binary score due to the deterministic nature of the techniques, for example any security technique can either achieve a parameter or not; cryptographic techniques, by definition, cannot be probabilistic. This configuration allows the comparison between FHE with the base and control cases of AES and no encryption respectively. Communications are also timed to estimate the feasibility of implementation in practical scenarios.

Configuration 1
MP_f, MS_f
This is the no-attack configuration, which serves as a benchmark for all models.

Model	Integrity	Confidentiality	Authenticity	Non-repudiation	Privacy Preservationon	Overall Time(s)
Plain Text	No	No	No	No	No	0.0001
Aes	No	Yes	No	No	No	0.0175
FHE	Yes	Yes	Yes	Yes	Yes	3.7241

Configuration 2

MP_t, MS_f

This attack configuration modifies the payload but leaves the signature intact.

Model	Integrity	Confidentiality	Authenticity	Non-repudiation	Privacy Preservationon	Time
Plain Text	No	No	No	No	No	0.0001 + Δt
Aes	No	Yes	No	No	No	0.0175 + Δt
FHE	No	Yes	No	No	Yes	3.7241 + Δt

Configuration 3

MP_f, MS_t

This attack configuration modifies the signature but leaves the payload intact.

Model	Integrity	Confidentiality	Authenticity	Non-repudiation	Privacy Preservationon	Time
Plain Text	No	No	No	No	No	0.0001 + Δt
Aes	No	Yes	No	No	No	0.0175 + Δt
FHE	No	Yes	No	No	Yes	3.7241 + Δt

Configuration 4

MP_t, MS_t

This attack configuration modifies both the signature and the payload.

Model	Integrity	Confidentiality	Authenticity	Non-repudiation	Privacy Preservationon	Time
Plain Text	No	No	No	No	No	0.0001 + Δt
Aes	No	Yes	No	No	No	0.0175 + Δt
FHE	Yes	Yes	Yes	No	Yes	3.7241 + Δt

6.1 LIMITATION

It is clear from the results laid out above that FHE beats the baseline models in nearly all cases. However there are severe limitations to the design of the experiment. For example, without digital signatures, integrity would be compromised in all cases, as the use of FHE alone cannot ensure integrity of the data, but only confidentiality. Moreover, in the absence of any appropriate publicly available V2I dataset, it is difficult to estimate how this technique would perform in practice. A simplifying assumption that the cloud storage of V2I data would require the most minimal FHE operations such as addition and multiplication was used. In the case of more complex data analytics, it is expected that FHE would be much worse. The limitations of the design are chiefly as follows:

1. Lack of real-world data to dispatch actual operations
2. Lack of on-network adversaries that also include attack types such as packet drops
3. Lack of statistically rigorous control experiment and setup

The last limitation is hard to quantify precisely in the domain of Network Security in general. Cryptographic techniques are designed with security parameters in mind which are either achieved or not achieved. Running the same simulation and infinitum will rarely yield different results. In general, security parameters are subjectively qualified and treated as a qualitative measure of the soundness of a technique.

A more thorough and well-funded design of the experiment would include the provision of a proprietary V2I dataset and the cloud computing resources necessary to replicate on-ground peculiarities. FHE operations are expensive and performing FHE operations on commercial compute machines is simply intractable for experimentation or practical purposes. However, the experiment was designed around the limitations to prove that FHE can be a viable solution to confidentiality woes in V2I communication by simplifying the client-server communication to a localhost setting and using a client-side attacker to replicate on-network attacks.

7 CONCLUSION AND FUTURE WORK

The purpose of this study was to establish the feasibility of homomorphic encryption to achieve security in a V2I setting. The objective of the study was to test the baseline improvements over traditional security techniques like AES or RSA encryption. Given severe limitations on the available data and compute resources, the experiment had to be carefully designed to replicate the V2I setting in a home environment. This study shows that, in theory, FHE beats the benchmark in most scenarios, coupled with digital signatures to ensure integrity. However even for the simple addition operation, FHE takes much longer than AES or other competitors. In a real-world V2I scenario where cloud storage must be computing data analytics for communication, we can expect this timing to be much worse. The true extent of this performance degradation can only be estimated in a more thorough and well-funded research endeavor with access to a real-world V2I dataset and lab compute resources. Future work on this research will investigate methods for speeding up computation for homomorphic encryption. High constraints are placed on homomorphic encryption operations; hence, the higher the constraints involved, the slower the scheme which leads to a longer computation time.

REFERENCES

Alaya, B., Laouamer, L. and Msilini, N. (2020) "Homomorphic encryption systems statement: Trends and challenges," *Computer Science Review*, 36, p. 100235. Available at: https://doi.org/10.1016/j.cosrev.2020.100235.

Amin, R. *et al.* (2018) "A light weight authentication protocol for IoT-enabled devices in distributed Cloud Computing environment," *Future Generation Computer Systems*, 78, pp. 1005–1019. Available at: https://doi.org/10.1016/j.future.2016.12.028.

Arora, A. and Kumar Yadav, S. (2018) *3 rd International Conference on Internet of Things and Connected Technologies, (ICIoTCT) 2018 Block chain Based Security Mechanism for Internet of Vehicles (IoV)*.

Baldini, G. *et al.* (2019) "Zone Keys Trust Management in Vehicular Networks based on Blockchain," *GIoTS, Global IoT Summit : 2020 conference proceedings*.

Benomarat, I., Madini, Z. and Zouine, Y. (2018) "Enhancing Internet of vehicles (IOVs) Performances Using Intelligent Cognitive Radio Principles," *2018 4th International Conference on Optimization and Applications (ICOA)*. IEEE.

Butt, T.A. *et al.* (2019) "Privacy Management in Social Internet of Vehicles: Review, Challenges and Blockchain Based Solutions," *IEEE Access*, 7, pp. 79694–79713. Available at: https://doi.org/10.1109/ACCESS.2019.2922236.

Cheng, W. *et al.* (2021) "A Survey on Privacy-security in Internet of Vehicles," in *2021 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech)*. IEEE, pp. 644–650. Available at: https://doi.org/10.1109/DASC-PICom-CBDCom-CyberSciTech52372.2021.00109.

Chim, T.W. *et al.* (2011) "SPECS: Secure and privacy enhancing communications schemes for VANETs," *Ad Hoc Networks*, 9(2), pp. 189–203. Available at: https://doi.org/10.1016/j.adhoc.2010.05.005.

Contreras-Castillo, J., Zeadally, S. and Guerrero-Ibanez, J.A. (2018) "Internet of Vehicles: Architecture, Protocols, and Security," *IEEE Internet of Things Journal*, 5(5), pp. 3701–3709. Available at: https://doi.org/10.1109/JIOT.2017.2690902.

Derdour, M. *et al.* (2019) "Vehicular Ad Hoc NETworks versus Internet ofVehicles - A Comparative View," *Proceedings, ICNAS 2019 : 4th International Conference on Networking and Advanced Systems : 26-27 June 2019*.

Fida, K. *et al.* (2019) "Cognitive Internet of Vehicles: Motivation, LayeredArchitecture and Security Issues*," 2019 International Conference on Sustainable Technologies for Industry 4.0 (STI) : 24-25 December, Dhaka.*

Gaid, M.L. and Salloum, S.A. (2021) "Homomorphic Encryption," in, pp. 634–642. Available at: https://doi.org/10.1007/978-3-030-76346-6_56.

Gentry, C. (2008) *Computing Arbitrary Functions of Encrypted Data*.
Han, S. *et al.* (2017) *2017 IEEE Global Communications Conference (GLOBECOM) : proceedings : Singapore, 4-8 December 2017.*

Henze, M. *et al.* (2016) "A comprehensive approach to privacy in the cloud-based Internet of Things," *Future Generation Computer Systems*, 56, pp. 701–718. Available at: https://doi.org/10.1016/j.future.2015.09.016.

Horng, S.J. *et al.* (2013) "B-SPECS+: Batch verification for secure pseudonymous authentication in VANET," *IEEE Transactions on Information Forensics and Security*, 8(11), pp. 1860–1875. Available at: https://doi.org/10.1109/TIFS.2013.2277471.

Huang, Q. *et al.* (2019) "Secure and Privacy-Preserving Warning Message Dissemination in Cloud-Assisted Internet of Vehicles," *2019 IEEE Conference on Communications and Network Security (CNS).*

Li, K. *et al.* (2020) "Efficient Message Authentication with Revocation Transparency Using Blockchain for Vehicular Networks," *Computers and Electrical Engineering*, 86. Available at: https://doi.org/10.1016/j.compeleceng.2020.106721.

Masood, A., Lakew, D.S. and Cho, S. (2020) "Security and Privacy Challenges in Connected Vehicular Cloud Computing," *IEEE Communications Surveys and Tutorials*, 22(4), pp. 2725–2764. Available at: https://doi.org/10.1109/COMST.2020.3012961.

Matsumoto, M. and Oguchi, M. (2021) "Speeding up Encryption on IoT Devices Using Homomorphic Encryption," in *Proceedings - 2021 IEEE International Conference on Smart Computing, SMARTCOMP 2021*. Institute of Electrical and Electronics Engineers Inc., pp. 270–275. Available at: https://doi.org/10.1109/SMARTCOMP52413.2021.00059.

Mohan, M., Kavitha Devi, M.K. and Prakash, J. v (2017) *Homomorphic Encryption-State of the Art*.

Mohiuddin, I. and Almogren, A. (2019) "Workload aware VM consolidation method in edge/cloud computing for IoT applications," *Journal of Parallel and Distributed Computing*, 123, pp. 204–214. Available at: https://doi.org/10.1016/j.jpdc.2018.09.011.

Mohiuddin, I. and Almogren, A. (2020) "Security Challenges and Strategies for the IoT in Cloud Computing," in *2020 11th International Conference on Information and*

Communication Systems, ICICS 2020. Institute of Electrical and Electronics Engineers Inc., pp. 367–372. Available at: https://doi.org/10.1109/ICICS49469.2020.239563.

Noh, J., Jeon, S. and Cho, S. (2020) "Distributed Blockchain-Based Message Authentication Scheme for Connected Vehicles," *Electronics*, 9(1), p. 74. Available at: https://doi.org/10.3390/electronics9010074.

Ramesh, S. and Govindarasu, M. (2020) "An Efficient Framework for Privacy-Preserving Computations on Encrypted IoT Data," *IEEE Internet of Things Journal*, 7(9), pp. 8700–8708. Available at: https://doi.org/10.1109/JIOT.2020.2998109.
Rasheed Lone, F., Kumar Verma, H. and Pal Sharma, K. (2021) "Evolution of VANETS to IoV," *Tehnički glasnik*, 15(1), pp. 143–149. Available at: https://doi.org/10.31803/tg-20210205104516.

Samara, G., Al-Salihy, W.A. and Sures, R. (2010) *Security Issues and Challenges of Vehicular Ad Hoc Networks (VANET) ghassan@nav6.org, 3 sures@nav6.org, 4th International Conference on New Trends in Information Science and Service Science*.

Stefan, R. and Slamanig, D. (2013) "Cryptography for Security and Privacy in Cloud Computing."

Tzeng, S.F. *et al.* (2017) "Enhancing Security and Privacy for Identity-Based Batch Verification Scheme in VANETs," *IEEE Transactions on Vehicular Technology*, 66(4), pp. 3235–3248. Available at: https://doi.org/10.1109/TVT.2015.2406877.

Xia, Y. *et al.* (2017) "Adaptive Multimedia Data Forwarding for Privacy Preservation in Vehicular Ad-Hoc Networks," *IEEE Transactions on Intelligent Transportation Systems*, 18(10), pp. 2629–2641. Available at: https://doi.org/10.1109/TITS.2017.2653103.

Yahiatene, Y. *et al.* (2019) "A blockchain-based framework to secure vehicular social networks," *Transactions on Emerging Telecommunications Technologies*, 30(8). Available at: https://doi.org/10.1002/ett.3650.

Zhou, X. *et al.* (2021) "Deep Correlation Mining Based on Hierarchical Hybrid Networks for Heterogeneous Big Data Recommendations," *IEEE Transactions on Computational Social Systems*, 8(1), pp. 171–178. Available at: https://doi.org/10.1109/TCSS.2020.2987846.