A MACHINE LEARNING BASED FRAMEWORK FOR SECURITY AGAINST DENIAL-OF-SERVICE IN VEHICULAR NETWORK INFRASTRUCTURE

Thesis Submitted for the Award of the Degree of

DOCTOR OF PHILOSOPHY

in

Computer Applications

by

Amandeep Verma

Registration Number: 41900012

Supervised by

Dr. Rahul Saha (18818)

Computer Science and Engineering (Professor)

LOVELY PROFESSIONAL UNIVERSITY, PUNJAB

2023

DECLARATION

I, hereby declare that the presented work in the thesis entitled "**A Machine Learning Based Framework For Security Against Denial-Of-Service In Vehicular Network Infrastructure**" in fulfillment of the degree of **Doctor of Philosophy (Ph. D.)** is the outcome of research work carried out by me under the supervision **Dr. Rahul Saha**, working as **Professor**, in the **Computer Science and Engineering Department/School** of Lovely Professional University, Punjab, India. In keeping with the general practice of reporting scientific observations, due acknowledgments have been made whenever the work described here has been based on the findings of other investigators. This work has not been submitted in part or full to any other University or Institute for the award of any degree.

(Signature of Scholar)
Amandeep Verma
Registration No. 41900012
Department of Computer Applications
Lovely Professional University, Punjab

CERTIFICATE

This is to certify that the work reported in the Ph.D. thesis entitled **"A Machine Learning Based Framework For Security Against Denial-Of-Service In Vehicular Network Infrastructure"** submitted in fulfillment of the requirement for the reward of degree of **Doctor of Philosophy (Ph.D.)** in **Computer Applications**, is a research work carried out by **Amandeep Verma**, Registration No. **41900012**, is a bonafide record of his original work carried out under my supervision and that no part of the thesis has been submitted for any other degree, diploma, or equivalent course.

(Signature of Supervisor)

Rahul Saha (18818)

Professor

Department of Computer Science and Engineering

Lovely Professional University, Punjab

ACKNOWLEDGEMENT

I am deeply grateful to my research advisor Dr. Rahul Saha for his motivating guidance throughout this research work. He provided me great freedom to choose a research topic of my interest and directed and critiqued me just in time. Without his inspiration, untiring efforts, enthusiastic attitude, and personal care, the accomplishment of this dissertation would have been impossible. His input was vital in shaping this work. He has been a great source of inspiration to me and I would like to thank him from the core of my heart. I gratefully acknowledge the School of Computer Applications, Lovely Professional University (Punjab), India for offering this doctorate program. The institute and the department were always very cooperative in extending all the resources and infrastructure facilities during my research span. I am thankful to all the reviewers of my research manuscripts for providing valuable feedback and comments that helped me to raise the standard of this research work. I also extend my special thanks to all the cited authors in my thesis whose research findings provide me with a battleground to choose from the research directions. Words are not adequate to express my gratitude to my wife Mrs. Yashwinder Kaur for always being there for me. She never failed to give me great encouragement and confidence so that I could finish my thesis. I am highly grateful to her for all the cooperation, affection, motivation, understanding, patience, and sincere advice. Her support and care helped me overcome setbacks and stay focused on my thesis. I greatly value his friendship and deeply appreciate her belief in me. I am also grateful to my father Mr. Om Parkash and my mother Mrs. Laxmi Devi for their blessings, best wishes, and words of encouragement. My mother cheered me up during my walk through the thorny journey towards attaining the Doctorate Degree. Also, I express my thanks to my other family members for always being with me at any stage during this work. Above all, I firmly believe that everything in the world is happening as per the plan and will of the Almighty. So, I thank God for planning this and allowing me to do this thesis. I pray that he continues showering his grace and blessings onto me and all others.

Amandeep Verma

Abstract

Vehicular Adhoc Network (VANET) is a modern implementation of an intelligent transport system that interconnects vehicles with each other and infrastructural modules. It belongs to the Mobile Ad hoc Networks (MANET) subclass, which inherits features from the Internet of Things (IoT). We use VANETs to inform drivers about traffic, tolls, and parking spaces and also use them for security, communication, business, and government surveillance. VANETs are expanding as a result of the growth of wireless networks. The development of Internet-based applications and dependency on these web-based applications results in the rapid increase of Internet-related attacks. Similarly, implementing security in VANETs and the upkeep of these networks have been challenging tasks. Among all the threats, Denial of Service (DoS) and Distributed Denial of Service (DDoS) attacks are more damaging. DDoS assaults have increased in volume and severity, making them a deadly tool for intruders since they may interfere with any network's functionality. All the legitimate users do not get the services they need. Since these assaults can result in economic damage for businesses, it is a crucial security concern. It may also risk human lives in vehicular networks. In vehicular networks, a DDoS attack, an extension of a DoS attack, can stop the functioning of the onboard GPS or entertainment systems; even the attacker can propagate a DDoS attack toward the electric grid through the charging stations for the connected electric vehicles. It is a challenging task to manage these attacks in a vehicular environment.

We use several Machine Learning (ML) methods for detecting DDoS assaults. ML finds the optimal solution for DDoS handling among all the available solutions. It is a subset of AI that allows machines to acquire knowledge and evolve without programmer intervention. Existing ML algorithms work as a single-layer security framework in a vehicular network environment. On the contrary DDoS dynamics are applicable in different network layers. Thus, a gap exists between the existing solutions and the multi-layer DDoS detection strategy requirements. Besides, most of the existing detection models need to consider the heterogeneity of the traffic and are non-adaptive as per the traffic rate. Organizations often face a strategic choice between prioritizing either DDoS prevention or detection due to resource constraints, operational considerations, and the dynamic nature of cyber threats.

We address the problem of DDoS attacks by introducing three security frameworks: one for DDoS detection, one for DDoS prevention, and a combined solution for both. For DDoS detection, we propose and use a security framework called *Vehicular Adaptive Intrusion Detection And Novel System for Heterogeneous Hosts (VAIDANSHH)*. It is the first ML-based Network Intrusion Detection System (NIDS) for DDoS attacks in vehicular networks to combine a three-tier security model, traffic adaptivity, and heterogeneity. These three tiers are the hardware tier (physical tier), the interface tier (communication channel), and the application tier. The advantage of this three-tier architecture is that it provides a safe vehicle environment by applying rigorous security at different levels. VAIDANSHH is a dynamic IDS that collects real-time packet information and adapts the thresholds of traffic according to the traffic load. This adaptation generates more accurate and reliable results. VAIDANSHH provides flexibility to incorporate vehicles of different vendors, standards, protocols, and technologies. Traffic is also heterogeneous in VANETs. As a result, the heterogeneity of vehicles does not create any problems related to compatibility or interoperability.

We propose the *”Predictive Risk Evaluation for Vehicular Infrastructure Resilience (PREVIR)”*. PREVIR is a new solution that combines the Logit approach (statistical analysis) with the LogitBoost approach (machine learning) to thwart DDoS assaults in vehicular networks. The logit model of PREVIR forecasts the likelihood of a packet being malicious, and the machine learning approach enhances PREVIR's performance via iteratively tuning the model's regular updates based on new traffic data. We conducted several trials with PREVIR. We use the NSL-KDD public dataset, the CIC-DDoS public dataset, and our NS3-generated dataset. Numerous attack types, such as UDP flood, TCP flood, mixed flooding, U2R, Probe, and R2L assaults, are analyzed by PREVIR. Its mathematical abilities give us the best results with little training and without over-fitting. It may be used for single and multi-class classification. The chance of an intrusion happening in a vehicle scenario is predicted by PREVIR using the logit model. Using packet characteristics, PREVIR calculates the likelihood that a new packet is harmful or not.

At last we introduce the first ML-based combined solution for the prevention and detection of DDoS attacks in vehicular networks. To be more specific, our proposed system is the first to consider the adaptive traffic threshold to generate the alarm for a suspicious amount of traffic flow in an Intrusion Detection Prevention System (IDPS). We call our proposed approach as *Protecting vEhicular neTworks against distRibuted deniAl of service attacKs (PETRAK).* PETRAK uses four functions: prevention, alarm, training, and detection. PETRAK's alarming system uses the flow parameters and activates the detection module. The detection module uses an ML model to detect malicious packets. The prevention system works in two modes: immediate and future. PETRAK uses logistic regression to identify incoming packets and signatures of malicious packets to prevent future attacks.

In terms of results, VAIDANSHH detects UDP flooding, a form of DDoS attack with 99.9% accuracy within a very short time. PREVIR can classify packets with an accuracy of 100% and 99.99% for the two different datasets respectively. Our model also shows an average sensitivity of 100% and an average specificity of 58.33%. The comparative analysis shows that PREVIR's efficiency is 20% better on average in the prevention of malicious packets in contrast to cutting-edge models. We observe that PETRAK shows an accuracy of more than 99%. The findings demonstrate PETRAK's effectiveness as a method for detecting and preventing DDoS assaults in VANETs. New strategy development benefits from a thorough examination of these threats and countermeasures. These attack detection and prevention system helps in saving financial resources and other network resources along with human lives. By using these solutions researchers and industry can provide a secure working environment to the end users and organizations.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Living in the information age means information plays an important role in our actions. We use networks to transfer information in many of our daily activities, such as setting a voice-activated reminder, browsing the news on a tablet, and preparing meals in a programmed oven. Besides utilizing technology to escape traffic, other tasks also include bill payments, learning, performing office duties, purchasing digitally, and using digital platforms. The primary purpose of networks is to share information, which is essential for our society to function. A network connects two or more computers to exchange data through rules or protocols. Network technology has changed significantly since its inception, and computing is shifting towards mobile technology. We can control electronic products through WiFi or Bluetooth networks instead of manually operating them. We connect various appliances through networks, and this area is becoming vast as more devices are joining these networks. Networks can be peer-to-peer, client-server, or other architectures and many options are available based on an organization's requirements due to significant technological advancements.

This chapter is separated into several subsections. The subsection 1.1 defines the relevant networks with their scope. The subsection 1.2 defines security and its requirements in all the networks. The subsection 1.2.1 provides the full description of mandatory security features, CIA and AAA security models, DoS attacks, and the transition of DoS to DDoS attacks. The subsection 1.3 defines VANET with its general architecture. It also defines the implementation of security services in VANET and the risk of DoS and DDoS in VANETs. The subsection 1.5 defines basic DDoS mitigation strategies. It defines general schemes like prevention, detection, response, and tolerance.

1.1 Networks categorization

According to Kizza, “a computer network is a distributed system consisting of loosely connected computers and other devices. These devices are referred to as network elements or transmitting elements, and any two can communicate through a communication medium” [1]. Each device must follow the rules or protocols to be in a communication network when communicating with another device. The combination of hardware and software that results from this is called a computer communication network or simply a computer network. We classify networks based on their architecture as follows.

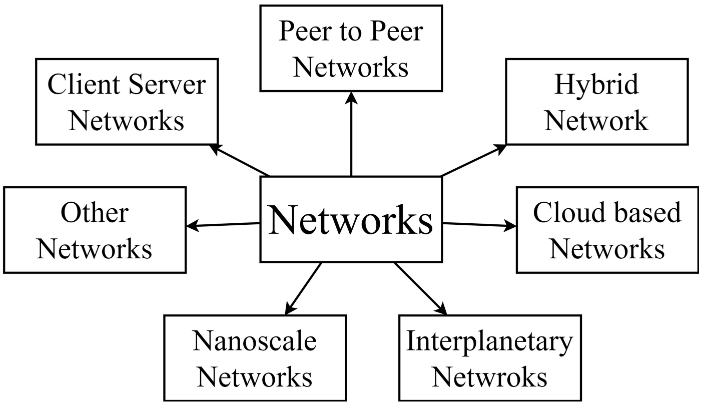

Figure 1: Networks categorisation

Client Server network: According to Migues, when a workstation can control a huge quantity of stored information and carry out sophisticated calculations, we use this network [2]. This system, which serves other systems, is referred to as a server. Servers come in a variety of forms. Client computers also access system resources simultaneously with the server. Clients use guided media, like wires, and unguided media, such as radio waves, microwaves, and satellites to access these server resources.

Peer-To-Peer network: A peer-to-peer (P2P) network is a computer network where every computer, or node, in the network can act as a client and a server. In other words, each computer can share its resources and access resources shared by other computers in the network. P2P networks are decentralized, which implies that the system is not controlled by a single server [3]. Instead, each computer communicates directly with other computers in the network. One of the advantages of a P2P network is that it allows for the easy sharing of resources, such as files or processing power, between computers in the network. This type of network is often used for file sharing, where users can download and upload files directly from and to other users in the network. P2P networks also provide a high level of fault tolerance, as the network can continue to function even if some nodes fail or are removed. Because of the absence of a central body to administer and safeguard the network, Peer-to-peer connections become prone to cyber threats like malware and viruses. Another potential issue with P2P networks is that they can be slower than client-server networks, as each computer must handle sending and receiving data. When we need resources spread out over several locations, we employ this network. We disperse them among different areas. With loose coupling, we link the various components of these networks together. It implies that no system is dependent on another system.

Hybrid network: A hybrid network is a type of computer network that combines two or more different network types or architectures. In a hybrid network, multiple servers provide resources to clients, but some of the clients can also act as servers and share resources with other clients. This type of network allows for a more flexible sharing of resources than a traditional client-server network while providing the security and management benefits of a server-based architecture [4]. One example of a hybrid network is a client-server network with a P2P overlay. In this type of network, multiple servers provide resources to clients, but the clients also connect directly to each other in a P2P overlay network to share resources. This strategy allows clients to access resources from both the servers and other clients in the network.

Cloud based networks: A cloud-based network is a computer network where the infrastructure and resources are hosted on cloud servers instead of being on-premises. Businesses and organizations often use these networks to provide access to applications, data storage, and other resources to users from anywhere in the world through the internet [5]. In a cloud-based network, the servers, storage devices, and other infrastructure components are hosted and maintained by a cloud services provider. Users can access these resources using a web browser or a dedicated software application, while the service provider of the cloud handles the network infrastructure's management, maintenance, and security. The concept involves the integration of all network management functionalities with cloud computing. Companies and institutions are migrating their processes to Internet facilities as technology progresses to save, transfer, and exchange information. This infrastructure is known as cloud technology. Nowadays, we see a wide range of devices connected to the internet using numerous sensors and other mechanisms. The Internet of Things (IoT) is a system of interconnected things that can carry out numerous functions including communication. This is referred to as the ”net of everything,” which is a more comprehensive phrase. To ensure proper operation of these networks, efficient data forwarding is essential, as data is often time-sensitive [6].

Interplanetary networks: Interplanetary Networks allow communication between planets and other celestial bodies in space. This type of network is designed to support deep space exploration, enabling communication and data exchange between spacecraft, orbiters, and ground stations across vast distances. Traditional networks, such as the Internet, are unsuitable for interplanetary communication due to the long distance and the time delay caused by the speed of light. The architecture of interplanetary networks relies on a store-and-forward approach, where data is transmitted from one node to another, and each node stores and forwards the data to the next node until it reaches its destination. This approach is necessary to account for intermittent connectivity and delays in signal propagation [7].

Nanoscale networks: Nanoscale networks refer to a type of network infrastructure that operates at the molecular and atomic scale. These networks are made up of nanoscale devices. These nanoscale networks have the potential to revolutionize various fields, including medicine, manufacturing, and computing. One of the challenges of nanoscale networks is the limited range of communication between devices. Due to the devices’ small size, communication is limited to short distances. Additionally, the devices are often susceptible to noise and interference from their environment, which can cause errors in communication. Despite these challenges, research in nanoscale networks continues to grow [8]. Such systems are employed in a variety of industries, including the manufacture of microchips and medicine.

Other networks: Many types of computer networks are commonly used in various settings.

The most well-known types include LANs, MANs, and WANs, which are often used to connect devices within a limited geographic area, a metropolitan area, or across large distances, respectively. However, several other popular networks serve unique purposes [9]. These include HAN, PAN, VPN, SAN, and GAN etc. Each of these network types has its features and implementations.

1.2 Network and security

Business organizations use a variety of networks. By utilizing these networks, businesses make themselves smart by increasing network speed, timely data exchange, and security. The networks have become faster and more scalable due to technological advancements. Still, security flaws are frequently increasing and decreasing the network's trustworthiness. Organizations need to ensure a safe working environment to gain customer trust, but in the cyber world, no one is entirely safe from potential threats. While people can take measures to protect themselves from existing threats, they cannot anticipate new ones that may emerge in the future. There have been instances where attackers have taken advantage of the vulnerabilities of the Internet. Therefore, it is essential to stay vigilant and stay ahead of attackers. The list of cyber threats is endless. It is imperative to respond to online society's safety concerns.

1.2.1 Security models (CIA and AAA)

The CIA model stands for Confidentiality, Integrity, and Availability. It is an important principle in the security field that strives to safeguard critical data privacy and confidentiality, preserve the accuracy of the data, and guarantee that authorized parties have permission to see its required contents. The AAA model, on the other hand, stands for Authentication, Authorization, and Accounting. It is a model used in access control systems, which verifies the identity of users (authentication), determines what actions they are allowed to perform (authorization), and keeps track of their actions for auditing purposes (accounting). While the AAA model helps to guarantee that only authorized people are provided information accessibility and that their activities are recorded for responsibility, the CIA model assists in guaranteeing that confidential data is secured against unauthorized access [10]. We illustrate the security model in Figure 2.

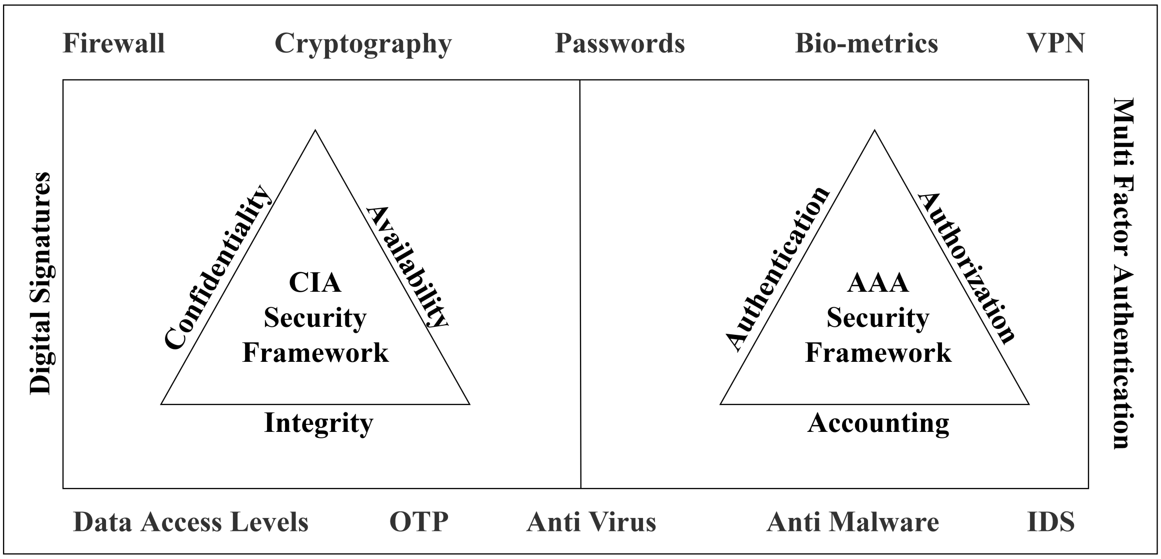

Figure 2: CIA Framework and AAA Framework

In general, security experts have successfully addressed many security issues, but they have mostly overlooked ”Availability.” However, due to the rising prevalence of DoS and DDoS attacks, this aspect has also become more critical, as users need access to information whenever needed. DDoS and Availability are the topics of this research.

1.2.2 Denial of Service (DoS)

A DoS (Denial-of-Service) attack is a type of cyber attack that aims to make a website, network, or online service unavailable to its intended users by overwhelming it with traffic or other types of requests [11]. The goal of a DoS attack is to prevent legitimate users from accessing the targeted system or service by flooding it with an excessive amount of traffic or data. This can cause the system to slow down or crash, rendering it unusable until the attack is mitigated. DoS attacks can be carried out in different ways, and their severity can range from minor inconveniences to significant disruptions or even catastrophic failures. There are several types of DoS attacks that attackers can use to achieve their goals, including volumetric attacks, application layer attacks, and protocol attacks. Volumetric attacks involve sending large amounts of data to the target system to consume its bandwidth and resources. Application layer attacks, exploit the system vulnerabilities to consume system resources and crash it. Protocol attacks, on the other hand, target the underlying communication protocols of a system, causing it to malfunction or crash [12].

1.2.3 Transition from DoS to DDoS

Computer processing speed, data retention, and connection speed have grown as technology advances. More security measures have been established to use the internet as a result of defending against conventional DoS assaults. Because systems now have more computing capacity, it is simpler to control these assaults, which has lessened their impact. But because attackers are always looking for new tactics, they have created a new technique called Distributed Denial of Service (DDoS), as seen in Figure 3.

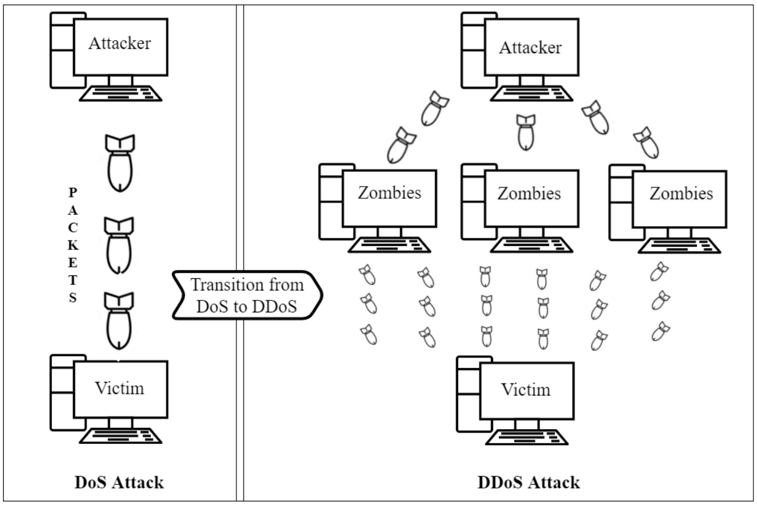

Figure 3: Transition from DoS attacks to DDoS Attacks

Figure 3 shows how a centrally controlled DoS assault transforms into a DDoS attack. The transition from DoS to DDoS attacks marks a significant evolution in the tactics attackers use to disrupt network services. In a DoS attack, a single computer floods a network or website with traffic, causing it to crash or become unavailable to users. However, with technological advancement, attackers have shifted to DDoS attacks that involve multiple computers working in tandem to overload the targeted network or service. Unlike DoS attacks, DDoS attacks are coordinated and can originate from anywhere in the world, making it difficult to trace the source of the attack. The transition from DoS to DDoS attacks has made it more challenging for network administrators to protect their networks and services from these malicious attacks. The attackers can use many compromised devices, making it challenging to filter out legitimate traffic from malicious traffic.

DDoS attacks use a coordinated collection of systems to overload the targeted network or service. Attackers can use bots or other infected machines to launch the attack directly or indirectly. To establish a DDoS attack, the attacker first identifies vulnerable devices that can be used as intermediate nodes. Attackers compromise these machines by injecting malicious code into them through various means, such as phishing emails or exploiting known vulnerabilities. Once the attacker injects the code, the compromised machine becomes part of a botnet to launch the attack [13]. The attacker then communicates with the compromised agents to inform them about the attack timing and the target. The attacker can use the botnet to flood the target network or service with massive traffic or exploit specific vulnerabilities in the target system. The attack aims to overwhelm the target and render it inaccessible to legitimate users. DDoS attacks launched using bots and injecting malicious code can be challenging to defend against, as they can originate from anywhere in the world and involve many compromised devices. Organizations can use DDoS protection services to detect and block malicious traffic, but attackers can use sophisticated techniques to bypass these defenses. Therefore, it is essential to implement a layered defense approach that includes regular vulnerability assessments, intrusion detection systems, and continuous monitoring to detect and respond to DDoS attacks in real-time.

1.3 VANET

The Internet has undergone significant changes in size, capabilities, technologies, and data traffic. Over time, the types of connections available to users have expanded from dial-up links to broadband and wireless connections such as 2G, 3G, 4G, and soon 5G. Researchers have achieved speeds of up to 45 terabits per second, allowing for connecting different devices and objects. These networks have sparked a new revolution, the Internet of Things (IoT) [14]. In 1999, Kevin Ashton, a UK-based entrepreneur and technologist coined the term IoT. He had developed standards for RFID technology [15]. In previous methods, physical objects or things could be connected to the internet using sensors or other technology, but they did not use internet protocols. A new approach has been adopted, which involves implementing IP on electronic devices so that they can communicate via the Internet. The first experiment involved an electronic toaster that turns on or off using the internet. Subsequently, IP connected other electronic devices, such as soda machines and coffee pots, to the internet. As a result, IoT has grown significantly in size. The expanding range of technologies in the Internet of Things (IoT) has increasing applicability, including transportation. We require an Intelligent Transport System (ITS) as we use millions of automobiles, railways, and airplanes. ITS is an architecture that uses various cutting-edge technologies to make the conventional transportation system safer, more practical, and blockage-free. it safer, more convenient, and congestion-free. VANETs have the same objective [16], facilitating communication between vehicles and other units. VANETs are established when vehicles connect with any infrastructure module [17]. We use them for various purposes, including commercial communication among vehicles, security, and surveillance by authorities. VANETs fall under the umbrella of MANETs and constitute part of the Internet of Things. We employ these networks in urban areas to inform drivers about congestion, tolls, and parking [16]. We use these networks to ensure passenger safety and comfort. Security concerns are also increasing with the growing scale of VANETs [17]. EVs are a crucial element of VANETs because they can communicate with both VANET vehicles and infrastructure, as well as with Smart Grids (SGs)[18]. The automobile industry has identified EVs as an important aspect of its future growth. We charge EVs through smart charging, which entails sharing data between the EV and charging equipment with the charging operator. Smart grids facilitate this type of electrical system[19]. Alternative advanced charging methods, such as vehicle-to-vehicle charging systems instead of intelligent grids, are also feasible, where vehicles meet at a designated location to exchange electricity [20]. This approach allows a vehicle to charge its battery without traveling far while the energy-providing vehicle also avoids excessive travel. There are multiple ways to describe the architecture of VANET, including component-based, communication technique-based, and hybrid approaches. VANET can include various types of vehicles, such as electric, non-electric, or hybrid vehicles, and can use different communication techniques, including cellular networks and dedicated short-range communication (DSRC). Component-based architecture involves various components, such as infrastructural components, mobile components, the Internet, and other related components. Communication-based architecture focuses on communication between vehicles rather than the network components used. In a hybrid approach, components and communication methodology are equally important. This method minimizes communication latency while boosting connectivity between cars and infrastructure thanks to the network's design. In the VANET design, we use various networking technological advances, including wireless networks and DSRC. It operates in the 5.9 GHz frequency band and transmits data up to a range of 300 meters. VANET architecture also uses cellular networks, which include 4G and 5G networks. A visual representation of the VANET is visible in Figure 4.

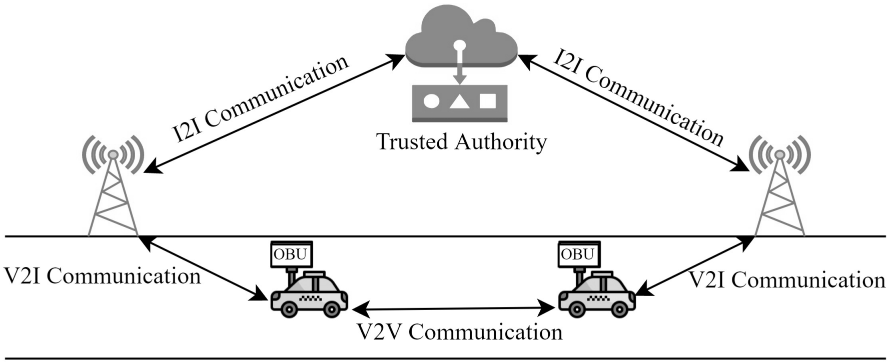

Figure 4: Architecture of vehicular ad hoc network

1.3.1 Security in VANET

As discussed in the previous section, various security requirements include confidentiality, integrity, and availability. Based on these services, the attackers may launch various VANET attacks. These attacks are shown in the Figure 5.

Among the numerous types of attacks that VANETs are vulnerable to, the DoS/DDoS attack is considered highly dangerous due to its universal nature, making it more potent than other types of attacks. Such attacks focus on the availability factor and cause a denial of service, thus rendering the services inaccessible. While there are other attacks as well, the focus will be on the DoS/DDoS attacks that threaten VANET users by disrupting the availability of services.

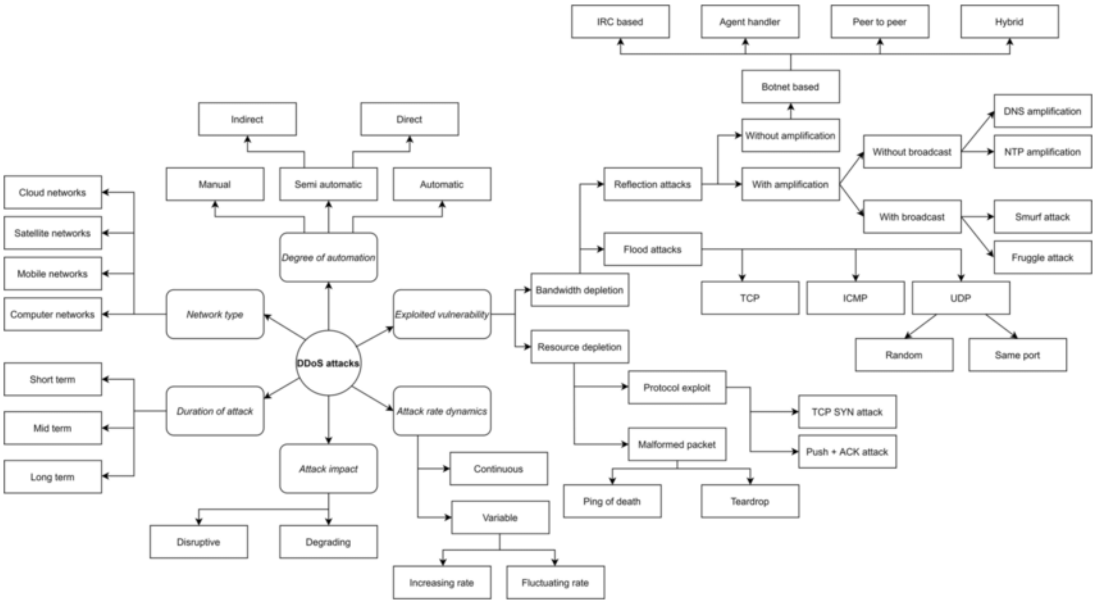

Figure 5: Attacks in VANET

1.3.2 DDoS in VANETs

DDoS attacks have become a significant concern in VANETs. A DDoS attack prevents an online system or application from being available by flooding it with many queries or streams of traffic, preventing it from responding to valid requests. DDoS attacks are difficult to mitigate, and they pose a significant threat to the reliability and safety of VANETs. In VANETs, DDoS attacks target various communication channels, including V2V and V2I communication links. The attacker can launch an attack with malicious vehicles or compromised devices. The attacker sends massive traffic to the targeted network, causing it to crash or become unresponsive. One of the primary reasons DDoS attacks harm VANETs is their impact on the availability of critical services, such as emergency messages, traffic management, and collision avoidance. These services rely on real-time and reliable communication among vehicles and infrastructure, and any disruption or delay in this communication can result in severe consequences, including accidents and fatalities. DDoS assaults also affect the effectiveness of VANETs by introducing an obstruction, worsening the connectivity's delays, and lowering its total performance. DDoS assaults pose a severe problem for VANETs, and as new attack routes keep developing, it becomes more challenging to identify and stop attacks. Therefore, more study is required to provide reliable and effective methods to reduce DDoS assaults in VANETs.

1.3.3 DoS/DDoS sub-types

Attackers can use any of the assaults, i.e., simple DoS assaults and extended DoS attacks. The attacker chooses a simple DoS assault when that attacker wants to deplete the resources of a specific node. These resources include a vehicle or any infrastructure unit. On the other hand, the attacker selects extended attacks that forward large volumes of data, disrupting communication between multiple infrastructure units and vehicles. These basic and extended attack types include different attack types, and Figure 6 illustrates some of these possible attacks in VANET.

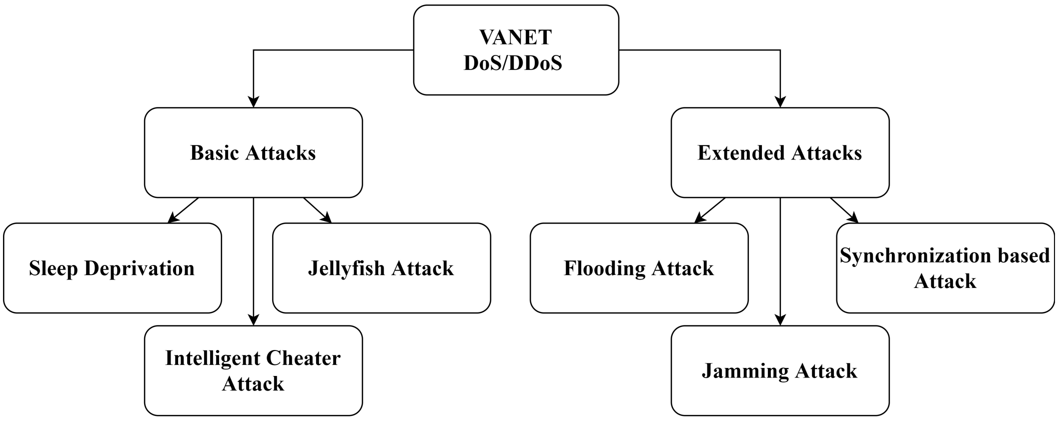

Figure 6: DoS/DDoS attacks in VANET

Sleep deprivation: A sleep deprivation attack attempts to make the victim's machine or communication system unavailable by making the target machine always active. The attack involves continuously sending packets or requests to the target system, preventing it from entering sleep or idle mode [21]. Continuous data transmission causes the system to consume more power, processing resources, and bandwidth than it normally would, leading to slow performance or complete system failure [22]. Sleep deprivation attacks can affect many systems, including servers, routers, and other devices. Since this kind of assault relies on saturating the system with traffic rather than depending on system flaws, it is complicated to protect against.

Flooding attack: Flooding attacks are a type of cyberattack in which the attacker sends an enormous amount of falsified data packets toward the victim. These packets become overwhelming for the victim due to their limited resources. The primary aim of the attacker in such an attack is to send an enormous amount of data, and therefore they send all these packets using fake or spoofed IDs [21]. As the attacker uses such IDs, no node or vehicle responds to them by sending RREP packets. Once the attacker enters the network, they establish a path of connections between nodes, which is used to transmit a vast amount of fake traffic. This traffic floods the network and consumes its available bandwidth, disrupting communication between various nodes.

Synchronization-based DDoS attacks: The synchronization-based DDoS attacks involve sending bogus packets at a specific time when legitimate service messages are being transmitted. As a result, the fake and legitimate messages collide, and the nodes in the network cannot receive legitimate messages. These attacks can be even more destructive when multiple attackers simultaneously send false messages. This type of attack causes significant disruption to the network's functionality and is challenging to detect and mitigate. The research conducted by Biswas and colleagues highlights the dangers posed by these attacks and the need for effective countermeasures to ensure the security and reliability of VANETs. [23].

Jamming attacks: Jamming attacks in VANET refer to deliberately interfering with wireless communications in the network by obstructing the communication frequency band. The attacker sends high-power signals on the same frequency band as the VANET, causing interference and blocking legitimate transmission [22]. This attack aims to disrupt communication between honest nodes and make the network unavailable to users. These are divided into 4 categories: constant, deceptive, random, and reactive [24]. In constant jamming, the attacker continuously transmits high-power signals, whereas in reactive jamming, the attacker selectively jams the channel only when a particular message is sent. Reactive jamming is more challenging to detect than constant jamming because it does not produce a continuous signal. Deceptive jamming, also known as signal masking, involves the transmission of RF signals that mimic legitimate messages to interfere with the regular communication between nodes. Random jamming involves the transmission of RF signals at random intervals.

Jellyfish attack: A DDoS assault known as a ”jellyfish” occurs when an attacker enters the infrastructure and blocks the transmission of data that has been received. By introducing delays in forwarding, the attacker causes packet drops, leading to the unavailability of packets to the intended recipients. Essentially, the attack causes network congestion by slowing packet transmission, which can fail communication between network nodes [21] [25].

Intelligent cheater attack: Intelligent cheater attacks in VANETs are a type of security threat that targets the system's trustworthiness. In these attacks, the malicious node tries to deceive other nodes by creating a false sense of reliability. The attacker can use a variety of tactics, such as sending incorrect information or modifying the data transmitted to other nodes. The objective of an intelligent cheater assault is to outperform other network nodes. The invading node's behavior appears normal, so such assaults might be challenging to identify. As such, there is a need for advanced security measures that can identify and prevent these attacks [25].

1.4 Machine Learning

Machine learning is a subset of artificial intelligence (AI) that focuses on the development of algorithms and models that enable computer systems to learn and make predictions or decisions without being explicitly programmed. In other words, it involves the use of data to teach a computer system to recognize patterns, make inferences, and improve its performance on a specific task over time. Key characteristics of machine learning include:

Learning from Data: Machine learning algorithms are designed to learn from large volumes of data. The more data they have access to, the better they can learn and make accurate predictions or decisions.

Pattern Recognition: Machine learning algorithms excel at recognizing patterns and relationships within data. They can identify complex patterns that may be difficult for humans to discern.

Generalization: ML models aim to generalize from the data they've been trained on to make predictions or decisions on new, unseen data. This ability to generalize is a crucial aspect of machine learning.

Adaptability: ML models can adapt and improve their performance over time as they are exposed to more data and gain experience. This adaptability is often referred to as ”learning.”

Automation: Machine learning allows for the automation of tasks that typically require human intelligence, such as image recognition, natural language processing, and decision-making.

Machine learning can be categorized into various types, including supervised learning (where models learn from labeled data), unsupervised learning (where models discover patterns in unlabeled data), and reinforcement learning, where models learn through trial and error based on rewards or penalties. Machine learning has a wide range of applications, including in fields such as image and speech recognition, recommendation systems, autonomous vehicles, medical diagnosis, fraud detection, and many others. It is a rapidly evolving field with ongoing research and development, and its applications continue to expand across various industries.

Machine learning significantly enhances security in VANETs by providing intelligent mechanisms for anomaly detection, intrusion prevention, and behavior analysis. ML algorithms can continuously monitor and analyze the vast and dynamic data generated by VANETs, enabling the early detection of security threats, such as unauthorized access, malicious activities, and network intrusions. By learning from historical data patterns, ML models can identify anomalies in real time, allowing for swift responses to mitigate potential risks and ensuring the integrity, privacy, and safety of vehicular communications. Additionally, the adaptive nature of ML enables VANET security measures to evolve and stay resilient in the face of evolving cyber threats, making it an indispensable component in safeguarding the reliability and trustworthiness of VANET infrastructure and services.

1.5 DDoS defense mechanism

DDoS defense strategies can be categorized in a variety of ways. It may be categorized based on the procedures employed or the place of implementation. Defense systems function in four phases for all assault types: preventing, detecting, reacting, and endurance and mitigating. The DDoS defense system is comprised of these four steps, as depicted in Figure 7.

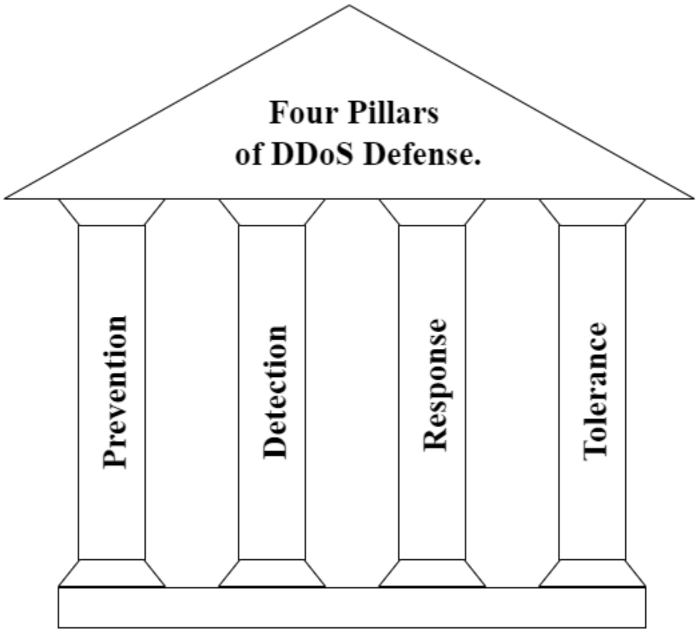

Figure 7: DoS/DDoS defense mechanism

A DDoS defense system's initial line of defense is to stop any attacker from starting an attack. If organizations can thwart an assault before it is launched, they will incur minimal costs. Before an attack is launched, we spot unusual behaviors that may be the start of such an attack and take the necessary precautions. We may employ globally synchronized filtering to prevent packets from creating significant attacking spikes. Filters that are globally synchronized collaborate with other gadgets that have Internet access. Besides these filters, we may utilize different additional attack prevention strategies for security [26].

Although DDoS preventive measures are constantly proactively engaged, given the progressive nature of assault methods, surveillance systems should be sufficiently secure to quickly and effectively identify an assault if the attacker can execute DoS or DDoS [27]. Early detection of an assault means the victim will sustain minor damage and have the shortest recovery time. Delays in the attack's identification might have a lot of negative consequences.

We employ intrusion detection systems (IDSs) for this reason, which can identify assaults using a variety of tactics. Although we use IDS in various forms, its fundamental purposes always stay the same. In all types of IDS, controlling, tracking, identifying, and warning are some of its primary functions.

After detection, the subsequent stage is to react to the assault [28]. Our detection system has identified and picked up the incident once we've determined that the attacker has started it. The network manager must act right once to identify the source of the assault and neutralize it. Let's say the onslaught continues for a while. If it happens, the assailant will either exhaust the available bandwidth capacity or overload the target system, making it difficult to halt the assault. It is challenging to identify the offender simply by looking at the source IP attribute of the message since offenders consistently utilize faked IP numbers. The first stage in an assault's anticipation is a traceback. Through the procedure of "traceback," we may locate the perpetrator's origin by employing a variety of accessible methods. Once we've located the source of the assault, we may either use rate-limiting strategies or, if the assault continues to be active, discard every single packet originating from this IP address. DDoS assaults are impractical to eliminate; hence, keeping the service operational for legitimate users while the breach occurs is essential. We do so by employing mitigating and assault endurance strategies. We do this using two fundamental techniques: tolerating faults and quality of service. The additional methods include throttle and pushback architecture. We achieve attack detection and prevention using IDS and IPS which are discussed as follows:

1.5.1 Intrusion Detection Systems (IDS)

Intrusion Detection Systems (IDS) are fundamental to modern network security, serving as vigilant sentinels in the digital realm. IDS solutions, which include Network-based IDS (NIDS) [29] and Host-based IDS (HIDS) [30], [31], are designed to monitor network traffic, system logs, and user activities in real-time. Their primary purpose is to detect patterns or anomalies indicative of unauthorized or malicious behavior. NIDS scrutinizes network traffic at critical junctures, while HIDS focuses on individual hosts, providing in-depth insights into system-level activities. IDS functions encompass anomaly detection, signature-based detection, and real-time alerting, enabling prompt responses to potential threats. Effective deployment entails strategic sensor placement, continuous tuning, and seamless integration with other security components, enhancing network security through the early detection of suspicious activities.

1.5.2 Intrusion Prevention Systems (IPS)

Intrusion Prevention Systems (IPS) represent the vanguard of network security, building upon the foundation laid by IDS. IPS not only identifies threats but also takes proactive measures to prevent or block malicious activities in real time. This advanced security solution conducts deep packet inspection, analyzing the content of packets as they traverse the network. Deployed inline, IPS intercepts and evaluates traffic before it reaches its intended destination, enabling immediate threat response. Key functions of IPS encompass threat prevention, deep packet inspection, and inline deployment. Careful rule configuration is essential to minimize false positives, ensuring that legitimate traffic remains uninterrupted. Moreover, IPS systems require automatic updates to stay current with evolving threat landscapes, bolstering the security posture of organizations in the face of rapidly changing cyber threats.

1.6 Organization of Thesis

We organize the thesis into a succession of chapters. In Chapter 1, we cover the concepts of VANETs and DDoS as a gateway to the domain of communications and security. Chapter 2 presents a thorough literary overview of VANETs, noting potential concerns. Chapter 3 provides a comprehensive literature review of existing VANET solutions. Chapter 4 presents the research problem, objectives, motivation, and methodology. We present the overall methodology followed by methodology to achieve individual objectives. Chapter 5 presents the first research contribution, the VAIDANSHH adaptive intrusion detection system for heterogeneous hosts. In contrast, Chapter 6 presents the second contribution, the PREVIR statistical model for detecting DDoS attacks in VANETs. Chapter 7 wraps up the thesis by outlining potential directions for future study, summarising the significant contributions, and considering their ramifications.

1.7 Conclusion

In conclusion, this first chapter has laid the groundwork by categorizing networks, emphasizing the crucial interplay between network architecture and security, and introducing essential security models such as CIA (Confidentiality, Integrity, Availability) and AAA (Authentication, Authorization, Accounting). We also delved into the spectrum of Denial of Service (DoS) attacks and their evolution into more potent Distributed Denial of Service (DDoS) attacks, highlighting their significance in modern networks. Transitioning to Vehicular Ad Hoc Networks (VANETs), we explored the unique security challenges they present and the emergence of DDoS attacks within this context, shedding light on the diverse subtypes of DoS and DDoS attacks tailored to the vehicular environment.

Our discussion on machine learning principles, encompassing learning from data, pattern recognition, generalization, adaptability, and automation, has paved the way for advanced security mechanisms. These principles hold the potential to enhance the detection and prevention of DDoS attacks, aligning with our overarching thesis goal of strengthening VANET security. Furthermore, this chapter introduced key DDoS defense mechanisms, with a particular focus on Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS), as integral components in our forthcoming framework. With this comprehensive foundation, we are well-prepared to embark on the subsequent chapters, where we will design, implement, and evaluate innovative security solutions aimed at fortifying the reliability and safety of vehicular networks in the face of evolving cyber threats.

Chapter 2

Vehicular Ad-hoc Networks

We thoroughly survey the literature on vehicular ad-hoc networks (VANETs) in this chapter. First, we give a general description of VANETs, including their design, uses, and characteristics. Following that, we go into some of the difficulties VANETs have to deal with, including mobility management, safety, and capacity. In addition, we look at how security affects VANET throughput. Additionally, we carefully examine security in VANETs and emphasize different security needs. This section categorizes several attack types according to the perpetrator, VANET levels, parts, and EVs. We wrap off this chapter with a recap of the significant discoveries and revelations from the empirical study.

2.1 Introduction

VANETs are a subset of MANETs, which, in turn, are a subset of IoT. VANETs are a specific type of wireless network that falls under the umbrella of IoT. They share some common characteristics with other IoT networks, such as sensors, data exchange, and network connectivity. However, VANETs are more specialized and focus on vehicular communication, distinguishing them from other types of wireless networks. The main objective of VANETs is to provide road safety and reduce traffic congestion. They achieve this by enabling vehicles to exchange real-time information like speed, position, and direction.

Urbanisation is a global trend that makes cities more populated, which causes serious roadblocks, delays, and various travel-related problems. This urbanization trend is anticipated to persist. It is crucial to install transportation technology, such as VANETs, that increase the efficacy and security of the transit system to manage the increasing traffic. The VANETs deal with various urbanization-related concerns, such as easing traffic congestion and boosting traffic flow, consequently raising the standard lifestyle for those who reside in urban regions. By promoting the use of green automobiles like autonomous and electrically powered vehicles, VANETs also assist in lowering the environmental effect of transportation. Thus, VANETs offer a viable approach to controlling the issues brought on by urbanization and guaranteeing secure, effective, and environmentally friendly infrastructure for today and tomorrow.

The term ”Internet of Energy” (IoE) describes how numerous power-related equipment, like smart grids, clean energy sources, battery-powered automobiles, battery backups, and systems for handling energy, are connected. The IoE makes it possible for these items to communicate and exchange data with one another, resulting in an electrical landscape that is more effective, safe, and efficient [32]. When discussing EVs, VANETs and the IoE are connected. As EVs gain popularity as customers and energy suppliers, we expect they will play a bigger part in the energy system. By facilitating connectivity among EVs, the infrastructure for charging them, and the electrical grid, VANETs make it easier for Vehicles to be included in the electrical system. VANETs aid in optimizing charging procedures by giving current data on the position, availability, and pricing of charging points, as well as power consumption and supply. VANETs also aid in power preservation and transmission by allowing EVs to share power with fellow vehicles and the power grid, depending on power availability and demand [33]. Moreover, by providing safe connectivity and authentication between EVs, charging facilities, and the power grid, VANETs improve the power network’s protection, anonymity, and robustness. By offering continuous surveillance and analysis of the network traffic and behavior, VANETs also aid in detecting and mitigating assaults and other vulnerabilities to the electrical system.

Intelligent Transport Systems (ITS) are essential because they integrate various cutting-edge technologies to make the traditional transportation system smarter, safer, more practical, and congested-free [34]. Similar objectives led to the development of VANETs, which allow interaction among vehicles and other components [35, 36]. With the use of RSUs, VANETs first permit vehicle-to-vehicle communication, but over time, this communication expands to link people walking and infrastructure. Such systems are currently employed in big cities to transmit data on mishaps, barriers, parking spaces, and power stations. Establishing connectivity between all vehicles and the rest of the world, especially the Internet, which may link diverse applications with millions of people, is the core principle underlying the creation of these networks. We depict the connection between VANET and IoT in Figure 8.

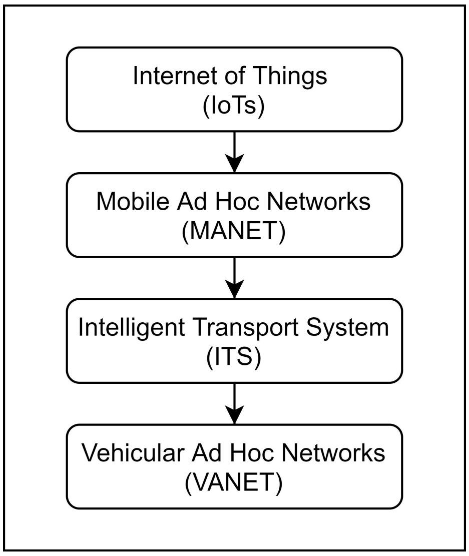

Figure 8: Relationship of Internet of Things (IoT) and VANETs

The Figure 8 shows that VANET is an instance of IoTs and benefits from IoT and MANET characteristics. The Internet of Things, MANET, ITS, and VANET, are all related technologies that aim to enhance various facets of the transportation sector. IoT is a networked equipment system that talks to one another to give people streamlined services. IoT may link cars to one another and the surrounding infrastructure in the transportation sector, enabling them to exchange data and improve performance. A wireless ad hoc network called a MANET enables devices to connect without a permanent infrastructure or centralized management. Regarding transportation, MANET may link cars to the local infrastructure, enabling users to exchange data and operate together on projects such as traffic management. ITS is an architecture that integrates numerous cutting-edge technologies into the traditional transportation system to make it more intelligent, safe, practical, and congested-free. It involves enhancing transportation networks utilizing IoT, MANET, and other technologies. A form of MANET called a VANET was created to facilitate communication among automobiles and other components in a vehicular environment. Because it offers the communication infrastructure required for ITS, VANET is a crucial part of ITS. IoT, MANET, ITS, and VANET, in short, are all interconnected technologies that enhance transportation systems by supplying real-time information, boosting safety, maximizing traffic flow, and allowing intelligent transportation systems.

2.2 Detailed architecture of VANET

VANET structure may be defined as either component-based or communication-based, among several other possibilities [37]. In the following section, we define the component-based architecture and then move on to defining the communication-based architecture. Before diving into these architectures, we illustrate a general scenario of VANET in Figure 9, which demonstrates a malware-based DDoS attack in the VANET with various components and parties involved in the network.

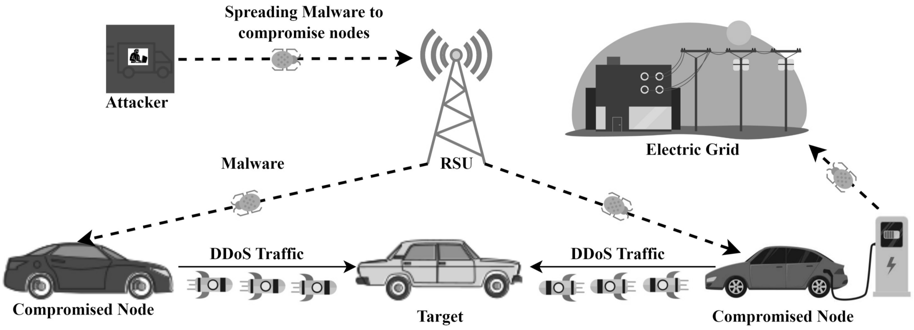

Figure 9: General architecture of VANET

2.2.1 Component-based Architecture

The foundation of a component-based VANET design is partitioning the network into discrete components or modules. Each component or module has a designated purpose, contributing to the network's overall functionality when these purposes integrate. This approach allows for greater flexibility and modularity in the design of the network, as well as easier maintenance and upgrading of individual components. OBU, vehicles, RSU, walkers, wireless networks, and charging infrastructures for battery-powered vehicles are all included in the VANET. Component-based architecture has the following components [38]:

Mobile components: These components include those that keep moving or remain in mobility within the VANET [39]. Mobile elements refer to the vehicles themselves, which are equipped with various communication devices such as GPS, sensors, and wireless communication interfaces. These devices transmit and receive data between vehicles and

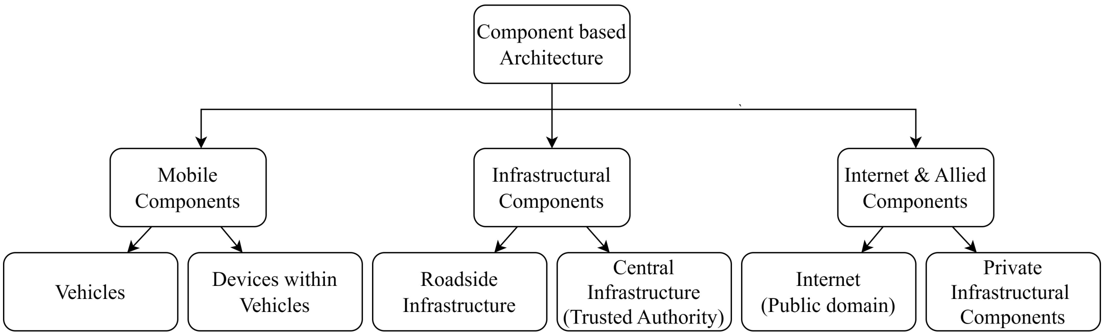
Component based Architecture
Mobile Components
Infrastructural Components
Internet & Allied Components
Vehicles
Devices within Vehicles
Roadside Infrastructure
Central Infrastructure (Trusted Authority)
Internet (Public domain)
Private Infrastructural Components

Figure 10: Component-based VANET architecture

between vehicles and infrastructure components. Mobile components also include handheld devices carried by passengers or drivers, which access information about traffic, weather, and other relevant data.

Vehicles: Since vehicles are the nodes that connect to create a wireless network, they are the essential movable part of VANETs. Vehicles are furnished with various connectivity methods to communicate with each other and infrastructural elements, such as other automobiles. Vehicles may communicate various information, such as traffic conditions, highway risks, temperature readings, and evacuation warnings, using VANETs as both information recipients and senders. Other vehicles can use this information to improve their judgments, such as lane changes, speed adjustments, or route selection. Depending on their functions inside the network, vehicles can also be categorized. As an illustration, certain automobiles may serve as routers or gateways, transferring information between other automobiles and infrastructural elements.

On-board units (OBUs): OBUs are essential VANET architecture components installed in vehicles to facilitate communication between vehicles and roadside units [40]. OBUs can be integrated into the vehicles during manufacturing or installed as an aftermarket device. OBUs collect and process data such as location, speed, acceleration, and vehicle heading. OBUs also receive information from other vehicles and RSUs, such as traffic congestion, accidents, weather conditions, and road work, and provide the same to the motorist. In addition to an accident, speeding restrictions, and lane deviation alerts, OBUs also alert the driver of potentially risky situations. OBUs typically consist of the following sub-components:

Wireless transceiver: The wireless transceiver enables communication between the OBUs and other OBUs or RSUs in the network.

GPS receiver: The position, acceleration, and trajectory of the vehicle are identified by the GPS receiver.

Antenna: The antenna sends and receives wireless signals.

Processor: The processor is in charge of handling the data that has been obtained and returning the pertinent information.

Memory: The memory stores the data that is being transmitted or received by the OBU. In addition, we use TPD [41], EDR [42], and GPS devices.

Infrastructural components: Infrastructural components refer to the fixed infrastructure deployed in the VANET environment. RSUs use various communication technologies such as Wi-Fi, Bluetooth, and cellular networks, which enable communication between vehicles and other infrastructure components [43] [44]. Other infrastructural components include traffic lights, surveillance cameras, and toll booths, providing connectivity and exchanging data with vehicles.

Roadside Units (RSUs): RSUs are fixed infrastructure components in VANET architecture installed along the roadways to facilitate communication between vehicles and other RSUs [45]. RSUs employ detectors and cameras to identify traffic jams, mishaps, adverse weather, and road construction to alert drivers and other RSUs in the network. RSUs can also receive information from other vehicles and RSUs to optimize traffic flow and provide real-time traffic information to drivers. RSUs can also communicate with traffic signal controllers to manage traffic flow and reduce congestion. RSUs comprise sub-components like wireless transceivers, antennae, processors, and memory.

Central trusted authority and server: Trusted authorities play a critical role in designing and operating secure and reliable VANETs. A trusted authority is responsible for managing the network's security and making sure that only permitted endpoints connect to the network, and transmissions are secure and protected from cyber threats [44]. The trusted authority also manages the distribution of information to vehicles and infrastructure nodes, ensuring that each device receives the information required to optimize its performance and enhance safety. In addition to managing security and information distribution, the trusted authority is also responsible for coordinating the VANET's operations. It includes collecting data flow between vehicles and infrastructure nodes, detecting traffic congestion, identifying potential hazards, and providing real-time traffic information to drivers. Overall, the role of the trusted authority is essential in creating a secure and reliable VANET ecosystem. Without a trusted authority, the network would be vulnerable to cyber-attacks, and data transmissions would be at risk of being intercepted or compromised. By managing security and information distribution and coordinating the VANET operations, the trusted authority ensures that the network functions effectively and that drivers and passengers can travel safely and efficiently.

The central server is a vital component of the VANET architecture responsible for managing and coordinating the VANET's operations. The central server receives data from OBUs and RSUs and processes this information to detect traffic congestion and identify potential hazards. The central server also manages the distribution of information to OBUs and RSUs, ensuring that each vehicle receives the necessary information to optimize its performance and enhance safety. The central server also manages the security of the VANET, ensuring that data transmissions are secure and protected from cyber threats.

Network Management System (NMS): The NMS is responsible for managing the entire VANET network. The NMS typically consists of many sub-components. The first is the network controller, which controls the overall network performance and coordinates the communication between the OBUs and RSUs. The security system is the second component responsible for ensuring the network's security and preventing unauthorized access or attacks. The traffic management system is the next sub-component accountable for managing traffic flow and reducing congestion. The last is a database that stores the network configuration and other network-related information.

Internet gateway: The Internet gateway is an essential component of the VANET architecture that connects the VANET with the Internet. The internet gateway enables OBUs and RSUs to access the internet, allowing them to communicate with networks and services outside the VANET, such as cloud-based traffic management systems, emergency services, and public transportation services. The internet gateway also enables remote management and monitoring of the VANET, allowing administrators to monitor network performance, diagnose issues, and update software and firmware.

Private infrastructural components: Private infrastructural components in VANET refer to the infrastructure owned and operated by private entities, such as toll road operators, private parking operators, and fleet operators. These entities may have RSUs, which communicate with their vehicles or customers. Private RSUs provide customized services to the vehicles that use their infrastructure. For example, toll road operators can use RSUs to collect tolls from passing vehicles and provide information on toll rates and payment options. Private parking operators can use RSUs to provide information on available parking spaces and reserve customer spots. Fleet operators can use RSUs to monitor the location and status of their vehicles, as well as communicate with drivers and provide navigation assistance. Private RSUs also augment the public RSU network. For example, private RSUs extend the range of the public RSU network, providing coverage in areas where the public network may not have coverage. Private RSUs also provide additional capacity and redundancy to the network, ensuring that the network handles large volumes of traffic and maintains connectivity even in the event of a failure of the public network. The use of private infrastructural components can improve the overall performance and reliability of the VANET, as well as provide customized services to users. However, the use of private infrastructural elements introduces challenges related to interoperability and coordination with the public network. Private infrastructure operators need to ensure that their components are compatible with the public network and adhere to VANET protocols and standards to ensure the smooth operation of the overall system.

Applications: Applications are responsible for providing services to users. Applications can be installed on OBUs or accessed through the internet gateway [40]. Applications offer various services, such as real-time traffic information, weather updates, parking availability, and emergency services. Applications can also provide customized services based on the user's preferences and driving habits, such as route planning, entertainment, and social networking. Applications facilitate the adoption of electric and automated vehicles, providing information on charging stations and autonomous driving capabilities.

2.2.2 Communication-based architecture

Communication-based VANET architecture refers to a type of VANET architecture where vehicular nodes transfer data with the help of RSU and directly. This architecture enables vehicles to share information, such as road conditions, traffic updates, and emergency alerts, to improve safety and efficiency on the road. Communication-based VANETs can be either infrastructure-based or infrastructure-less. In a framework built on fixed physical structures like terminals or gateways, vehicles may interact with one another. In contrast, in the infrastructure-less architecture, vehicles communicate directly with each other without the need for any fixed infrastructure.

The Communication system: Vehicles in VANET communicate using WAVE and DSRC [46]. VANET is unable to perform its basic jobs with information acquired through these methods. The Internet helps in getting a wide range of information. We use 3G/4G/LTE cellular networks to communicate. We categorize communication in different classes like in-vehicle communication [47] [48], V2V communication [49] [50], V2G communication [52] [53], V2P communication [51], V2B [54], V2I communication [50], and V2X [55] [56].

2.2.3 Cloud-based layered architecture

The layered architecture of VANET in the context of cloud computing in VANETs divides the overall architecture into four layers that include:

- Perception Layer: This layer gathers information from various sources such as sensors, cameras, and communication networks.
- Coordination Layer: This layer helps in coordinating various vehicle activities in the network and ensuring that they work together effectively.
- Artificial Intelligence Layer: It helps in making decisions based on the information gathered in the perception layer and coordinating the activities of vehicles through the coordination layer.

- Smart Application Layer: This layer provides various applications such as traffic management, safety warnings, and navigation services.

2.3 VANET applications

The way one makes use of VANET is specified by an application. We utilize these networks for various purposes concerning the types of interaction taking place among different groups. Such beings could engage in interactions between vehicles, vehicles, RSUs, or any other form of interaction [57]. The proposed applications include the following types:

Safety applications: VANETs improve safety on the road by providing real-time information about traffic conditions, road hazards, and potential accidents [58]. Here are some safety applications of VANET:

- Collision warning: VANETs enable vehicles to communicate with each other and exchange information about their speed, direction, and location. This information detects potential collisions and warns drivers to take evasive action.
- Emergency vehicle notification: Emergency vehicles, including medical, firefighting, and patrol cars, can use VANETs to alert neighboring drivers of imminent danger and ask them to make way.
- Traffic signal preemption: VANETs communicate with traffic signals and prioritize emergency vehicles, allowing them to reach their destination faster.
- Road hazard warning: VANETs enable vehicles to share information about road hazards such as potholes, construction zones, and debris on the road. This information warns other drivers and prevents accidents.
- Pedestrian safety: VANETs also improve pedestrian safety by detecting the presence of pedestrians and warning drivers to slow down or stop.

Overall, safety applications of VANET have the potential to reduce accidents, injuries, and fatalities on the road.

Comfort applications: Comfort applications of VANET improve the overall driving experience and make it more comfortable and convenient for the driver and passengers [59]. Some of the popular comfort applications of VANET include:

- Infotainment: VANETs provide infotainment services to drivers and passengers, such as music, news, and weather updates. These services are available through the in-car entertainment system, and we can customize them according to the driver's preferences.
- Navigation: VANETs also provide real-time navigation services to drivers, such as real-time traffic updates, road closures, and alternative routes. It can help drivers save time and avoid congested areas.
- Parking assistance: VANETs provide drivers with real-time information about parking spots in a particular area. It can help drivers save time and avoid unnecessary driving around searching for a parking spot.
- Remote vehicle monitoring: VANETs remotely monitor the status of a vehicle, such as fuel level, battery level, and tire pressure. It can help drivers plan their journeys more effectively and avoid unexpected breakdowns.
- Comfortable driving: VANETs also provide a more comfortable driving experience, automatically adjusting the car's temperature and humidity or the suspension to ensure a smoother ride.

Overall, the comfort applications of VANET make driving more enjoyable, convenient, and stress-free, ultimately enhancing the overall driving experience.

Commercial applications: Commercial applications of VANET are those that generate revenue or support business activities [60]. Some examples of commercial applications of VANET include:

- Location-based advertising: VANET sends location-based advertisements to drivers. Using data from the vehicle's OBU, companies can determine a driver's location and send promotions to nearby businesses.

- Fleet management: VANET tracks the location of a fleet of vehicles and optimizes routes to improve efficiency and reduce costs.
- Logistics and supply chain management: VANET tracks the location and condition of goods in transit, providing real-time visibility into the supply chain.
- Toll collection: VANET collects tolls automatically as vehicles pass through toll booths, eliminating the need for physical toll booths and reducing traffic congestion.
- Parking management: VANET manages parking in urban areas, directing drivers to available parking spaces and reducing the time spent looking for parking.
- Smart transportation systems: VANET can be integrated with other intelligent transportation systems, such as traffic management and public transportation systems, to provide a seamless and integrated transportation experience.

Environmental applications: Vehicular Ad-hoc Networks (VANETs) are wireless communication networks that enable vehicles to communicate with each other and with infrastructure in their vicinity [61]. VANETs have a wide range of environmental applications, including:

- Pollution monitoring: VANETs gather real-time data on air quality and other environmental factors, allowing authorities to monitor pollution levels and take appropriate measures to reduce emissions.
- Traffic management: By sharing information about traffic conditions in real-time, VANETs can help reduce congestion and improve traffic flow, reducing emissions and improving air quality.
- Intelligent transportation systems (ITS): VANETs can be integrated with ITS to provide drivers with real-time traffic information, helping them choose the most efficient routes and reduce fuel consumption.
- Green routing: VANETs provide drivers with the most environmentally friendly routes based on traffic conditions, terrain, and fuel efficiency.

- Eco-driving: By providing drivers with real-time feedback on their driving habits, VANETs can encourage eco-driving practices such as smooth acceleration and deceleration, which can reduce fuel consumption and emissions.

Overall, VANETs have the potential to play a significant role in reducing the environmental impact of transportation by enabling more efficient and sustainable practices.

Productive applications: The productive applications of VANET include benefits derived from its main advantages. One such benefit is the potential for VANET to help travelers save time, fuel, and money, and reduce environmental impact by providing shorter and more efficient travel paths [57]. VANETs optimize routes and reduce congestion by enabling vehicles to communicate with each other and the surrounding infrastructure, leading to faster and smoother travel. It leads to significant savings in fuel consumption and travel time for individual drivers, as well as reduced emissions and overall environmental impact. Therefore, the productive applications of VANET include optimizing traffic flow, reducing congestion, improving travel efficiency, and promoting sustainability.

2.4 Features of VANET

VANETs have gained significant attention recently due to their potential to improve road safety, reduce traffic congestion, and enhance transportation efficiency. However, VANETs also present unique challenges due to their highly dynamic nature, lack of fixed infrastructure, and attack vulnerability. In this context, understanding the features of VANETs is essential to designing efficient and secure communication protocols that can address these challenges. In this regard, some of the crucial attributes of VANETs include centralized security systems, time constraints, shared broadcast channels, volatility, no fixed topology, and infrastructure-less operation [62] [63]. Different characteristics of the VANET are as follows:

Centralized security system: Unlike traditional wireless networks, VANETs do not have a centralized security system responsible for implementing security among all nodes. Instead, servers implement security standards. As a result, ad hoc networks rarely monitor packets, which makes the network more vulnerable to attacks.

Time constraint: In VANETs, the nodes must forward messages within a specified time to avoid collisions in critical situations. However, verifying the authenticity of these messages can lead to extra delays. Because of this, VANETs put a time limit on nodes, requiring them to relay protected and legitimate data during a certain period.

Shared broadcast channel: VANETs send data through broadcasting, which makes it simple for attackers to steal the exchanged data. Due to these flaws, security in VANETs is a top priority.

Volatility and no fixed topology: VANETs are highly dynamic networks where vehicles frequently join and leave the network, making it challenging to maintain a network for an extended period. This volatility also makes it difficult to implement routing protocols due to the lack of fixed topology, so the security system frequently reconfigures the routes, which increases routing overheads.

Infrastructure-less: Ad hoc networks like VANETs are not dependent on routing devices or other stationary equipment. Through the use of reputation management systems, trust relationships between automobiles must be built in these infrastructure-free ecosystems.

2.5 Future of VANET

Urbanization is a global trend leading to an increase in population density in metropolitan areas, resulting in significant traffic congestion, delays, and other transportation-related issues. We expect this increase in urbanization to continue. By the year 2050, it is anticipated that 86 percent of the population in advanced nations and 64 percent of the population in emerging nations will live in urban areas. [64]. It is crucial to adopt VANETs that enhance the efficiency and safety of the transportation system to manage the increasing traffic. The deployment of VANETs addresses various issues related to urbanization, such as reducing congestion and enhancing traffic flow, thereby improving the quality of life for people living in urban areas.

EVs play a crucial role in reducing greenhouse gas emissions and providing a green environment in the future [65]. EVs run on electricity from renewable energy sources like solar, wind, hydro, or geothermal power. These energy sources significantly reduce carbon and other harmful emissions and reduce air pollution. Furthermore, EVs are more energy-efficient than conventional vehicles, requiring less energy to cover the same distance. The rising shift towards EVs also creates new jobs in the renewable energy industry and supports a green economy. The EVs also mitigate noise pollution as they operate quietly due to their electric motor technology. Therefore, deploying EVs provides a green environment by reducing emissions, saving energy, and reducing dependency on fossil fuels. The development and adoption of EVs and renewable energy technologies pave the way for a sustainable future that balances economic growth with environmental protection.

Several countries have implemented or are in the process of implementing EV policies. Gasoline stations are incorporating charging stations [66]. Additionally, businesses and EV owners install charging stations at their offices and homes. The emergence of intelligent metering technology enables the supply of electricity in both directions while tracking electricity consumption and distribution to the grid. We can give this surplus electricity back to the grid at a reasonable price, which motivates users to save more electricity and contribute to the grid, resulting in monetary benefits. The system requires enabling G2V and V2G electricity supply through smart meters. Researchers are considering some advanced versions of this technology where one electric vehicle (EV) can directly transfer surplus electricity to other EVs without the involvement of grids [67].

Vehicles and VANET infrastructure are vulnerable to security attacks [68]. Lack of security standards or limited use of this security standard makes VANETs insecure. Security attacks affect various VANET layers and disturb services like integrity, confidentiality, and availability [69]. Researchers suggest multiple methods to handle these attacks, but the frequency of these attacks is still rising. The attack rise is due to the incorporation of more electric and automated vehicles in VANET. Attackers launch attacks using unknown methods or with some variation in existing techniques, such as volume, the strategy used, or some other form. The following section discusses different security issues related to these attacks, along with various attack types and solutions. These subsequent sections also analyze different attacks and solutions.

2.6 Security requirements

We need security aspects and requirements for VANET (Vehicular Ad Hoc Network) to ensure the safety and security of the vehicles and passengers on the road. The transmission of details concerning the state of the highways and congestion, vehicle velocity, and other pertinent data is rendered through VANET, which increases road safety, lessens traffic jams, and boosts driving quality. However, several security risks affect vehicle and infrastructure communication, like envious attacks, data manipulation, and breaches of confidentiality. Security measures are crucial to prevent unauthorized access, safeguard data integrity and anonymity, and guarantee the network's secure and dependable functioning [69].

Confidentiality: Confidentiality refers to keeping sensitive information hidden from people who aren't authorized to access it while making it visible only to those who are. Confidentiality is essential for protecting data privacy and preventing unauthorized access to sensitive information.

Integrity: Integrity refers to ensuring that the information received by a recipient is the same as the information sent by the sender, without any unauthorized alterations. Integrity is essential for maintaining data accuracy and reliability and preventing tampering or corruption of information during transmission.

Availability: Availability is the ability for legitimate users to access the resources and services they need whenever needed. Ensuring availability is essential for preventing denial-of-service attacks and ensuring authorized users access the required information and services.

Non-repudiation: Non-repudiation is a service that guarantees the authenticity and integrity of the data transmitted and the legitimacy of its sender. Non-repudiation helps to prevent fraud and unauthorized access and ensures that data integrity is maintained.

Authentication: Authentication is the process of confirming a user's legitimacy and ensuring they have permission to obtain the data or facilities they're attempting to access. Authentication helps prevent unauthorized access and data breaches and ensures that only legitimate users can access sensitive information.

Authorization: Authorization defines the user access level to certain information or services. Authorization involves determining how much information a user can access, for how long, and what other services they are authorized to use. It also helps to prevent unauthorized access and data breaches and ensures that users can only access the information or services they are allowed to use.

Accounting: Accounting involves monitoring user activities and maintaining records of relevant statistics, such as the time a user has used resources and services. It helps to identify potential security threats and monitor system usage and is commonly implemented through log files.

ID traceability: ID traceability involves identifying the real identities of vehicles. ID traceability enables the correct source of a message to be located. These IDs are often employed to identify the actual sender and recipient, which helps to guarantee that conversations are routed and obtained from trusted parties [70].

Revocability: Revocability allows central authorities to revoke the certification and deregister a vehicle if it misbehaves, removing malicious nodes from the network. Revocability helps to maintain network security and prevent potential security threatsk [71].

Liability identification: Liability identification is based on non-repudiation services, which hold drivers accountable for any mistakes they may have made. It helps to ensure that drivers take responsibility for their actions and helps to prevent fraudulent behavior [72].

Real-time constraints: Real-time constraints require that vehicular information be delivered in real-time, with any delay potentially leading to severe consequences. Ensuring real-time information delivery is essential for maintaining system performance and preventing accidents or other security threats. Authentication is the process of verifying the identity of users and ensuring that they are authorized to access the information or services they are trying to access. Authentication helps prevent unauthorized access and data breaches and ensures that only legitimate users can access sensitive information.

2.7 Attack classification

Attack classification in VANETs can be categorized based on various factors such as security services, attacker type, VANET architecture layer, VANET component, and attacks on Electric Vehicles. These classifications help identify the attack's nature, its impact's severity, and its potential to harm the network. We can take appropriate measures to mitigate or prevent them by understanding the different types of attacks and their classifications. The classification of attacks based on security services includes attacks on integrity, availability, confidentiality, non-repudiation, and authentication. On the other hand, the category based on attacker type includes insider and outsider attacks. The classification based on the VANET architecture layer contains attacks on the network, physical, and application layers. Similarly, attacks on the various components of VANETs, such as RSUs, OBUs, and vehicles, can also be classified. Finally, the classification based on attacks on EVs includes attacks on the EV-charging infrastructure and the electric grid.

2.7.1 Attacks on security services:

Security services such as confidentiality, integrity, availability, authentication, and non-repudiation are crucial for safe and protected data transmission. However, attackers can exploit these security services' vulnerabilities and launch various attacks. For instance, attackers can launch man-in-the-middle, traffic analysis, social engineering, or eavesdropping attacks on confidentiality. Similarly, attacks on data integrity include masquerading, replay, message tampering, or illusion. Availability can be compromised by Denial of Service (DoS/DDoS), sleep deprivation, jamming, jellyfish, intelligent cheater, black-hole, grey-hole, greedy behavior, or spamming attacks. Authentication can be breached by Sybil, tunneling, GPS spoofing, free-riding, or certificate/critical replication attacks. Lastly, repudiation attacks or loss of events can target the non-repudiation service. Therefore, it is essential to understand these attack classifications and develop countermeasures to ensure secure communication in VANETs.

Confidentiality: The confidentiality of information exchanged between vehicles is crucial in VANETs. Different solutions, such as public keys and certificates, encrypt and make the information confidential. Despite these measures, attackers may launch attacks to breach information confidentiality through the latest attack methodologies. Security breaches that compromise confidentiality involve unauthorized access to or exposure to private data. These assaults aim to obtain information that should be kept confidential, such as proprietary knowledge, critical corporate information, or personal or financial data. Various attacks that disrupt the confidentiality services are as follows:

Man-in-the-middle attack: This attack type focuses on compromising the confidentiality of communication in VANETs. It occurs when an unauthorized entity intercepts communication between vehicles, allowing the attacker to eavesdrop on sensitive information or manipulate data in transit. Such attacks can lead to the exposure of private information and pose a threat to the privacy of vehicle users.

Traffic analysis attack: Traffic analysis attacks aim to breach confidentiality by analyzing patterns in network traffic. Attackers use this method to gain insights into the behavior of vehicles and their users, potentially revealing sensitive information. Protecting against traffic analysis attacks is crucial to safeguard the confidentiality of VANET data.

Social attack: Social attacks exploit human psychology to deceive vehicle users and obtain confidential information. These attacks may involve tactics like phishing or impersonation, where attackers trick users into revealing sensitive data. Maintaining confidentiality in VANETs requires defenses against social engineering attacks.

Eavesdropping attack: Eavesdropping attacks involve unauthorized parties listening in on the communication between vehicles, which can lead to a breach of confidentiality. Attackers eavesdrop to gain information they should not have access to, making it essential to secure VANET communication channels against such threats.

Data integrity: Integrity helps to guarantee that information that is exchanged isn't altered, rescheduled, or removed while being transmitted. An attempt to alter or destroy information without authorization is the aim of a data integrity attack. The attacker tries to change the accuracy or contents of the data, which might make it useless, unstable, or potentially harmful. Attacks that disrupt the integrity of services include the following:

Masquerading attack: Integrity attacks involve impersonation of legitimate vehicles to gain unauthorized access or manipulate network traffic. This type of attack threatens the integrity of data and trust within the VANET environment.

Replay attack: Replay attacks capture and retransmit data to deceive recipients into accepting false information, compromising data integrity. This can lead to incorrect actions or decisions being made by vehicles based on manipulated data.

Message tampering attack: Message tampering attacks involve altering messages in transit to corrupt data or deceive vehicle systems, undermining the integrity of the information exchanged in VANETs.

Illusion attack: Illusion attacks create deceptive signals to mislead vehicles' perception and decision-making processes, posing a significant threat to the integrity of the VANET environment.

Availability: An information security assault, known as an availability attack, aims to interrupt or limit access to an infrastructure or system so that it is inaccessible to those expected to use it. Availability is critical in VANETs as it ensures that all information is available to legitimate users when required. By flooding the system or network with data or taking advantage of flaws to bring about a system crash or failure, the attacker seeks to prohibit authorized users from accessing the system or network.

DoS/DDoS: Denial of Service (DoS) and Distributed Denial of Service (DDoS) attacks overwhelm the VANET network or specific vehicles with traffic, rendering them unavailable for legitimate communication and coordination. Ensuring availability in the presence of these attacks is critical for VANET functionality.

Sleep deprivation: Sleep deprivation attacks force vehicles into sleep mode, reducing their availability for communication and coordination. This can disrupt the responsiveness of vehicles in the network.

Jamming attacks: Jamming attacks emit radio interference to disrupt communication between vehicles, leading to unavailability. Effective countermeasures are necessary to mitigate the impact of jamming attacks.

Jellyfish attack: Jellyfish attacks flood the network with bogus messages, causing congestion and impairing service availability. Protecting against such attacks is crucial to maintain network efficiency.

Intelligent cheater attack: These attacks involve vehicles pretending to be cooperative while misusing network resources, negatively impacting availability by introducing deceptive elements into the network.

Blackhole attack: Blackhole attacks intercept and drop data, making it inaccessible to other vehicles or systems, severely affecting availability and data integrity.

Grayhole attack: Grayhole attacks partially drop data, leading to inconsistencies and reduced network availability, making it challenging to ensure data integrity and system availability.

Greedy behavior attack: Greedy behavior attacks involve selfish or malicious behavior that exploits network resources, negatively impacting availability and cooperation within the VANET environment.

Spamming attack: Spamming attacks flood the network with irrelevant data, consuming resources and degrading availability, which can hinder critical communication in VANETs.

Authentication: An authentication assault is a security assault that aims to obtain unauthorized access to a system or asset by evading or abusing its authentication measures. Validating a user's or system's identification through authentication is a procedure that frequently involves using credentials like usernames and passwords.

Sybil attack: Sybil attacks involve creating multiple fake identities to gain trust and disrupt network operations, posing a severe threat to the authenticity and trustworthiness of interactions in VANETs.

Tunnelling attack: Tunnelling attacks divert network traffic through unauthorized routes, compromising the authenticity of communication paths and potentially facilitating further attacks.

GPS spoofing: GPS spoofing attacks manipulate GPS signals to deceive vehicles about their physical location, impacting the authenticity of location-based services and navigation within VANETs.

Free-riding attack: Free-riding attacks exploit network resources without contributing, violating the authenticity of cooperation within the VANET environment, and potentially degrading the quality of service.

Certificate/key replication attack: These attacks involve illegally copying and using certificates or keys to gain unauthorized access, undermining the authenticity and security of the VANET infrastructure.

Non-repudiation: Non-repudiation attacks are a form of security attack designed to compromise the trustworthiness of digital transactions by enabling one side to free themselves of responsibility. The capability to assure that the parties engaged in a digital transaction cannot deny their involvement in it is known as non-repudiation. It ensures that once a particular message is sent, the sender cannot deny having sent it. However, attackers may launch repudiation attacks or cause a loss of events, leading to various security breaches. Therefore, non-repudiation is crucial in ensuring the authenticity and integrity of information transmitted in VANETs. Table 1 shows attacks launched on various security services.

Table 1: Attack classification based on security services

Attacked Service	Type of Attack	Reference(s)
Confidentiality	Man-in-the-middle attack	Ahmad et al. (2018) [73], Li et al. (2012) [74]
	Traffic analysis attack	Cencioni et al. (2008) [75]
	Social attack	Sumra et al. (2011) [76]
	Eavesdropping attack	Choudhari et al. (2019) [77]
Integrity	Masquerading attack	Malhi et al. (2016) [78]
	Replay attack	Junaid et al. (2018) [79], Malik et al. (2019) [80]
	Message tampering attack	Singh & Sharma (2019) [81]
	Illusion attack	Lo & Tsai (2007) [82]
Availability	DoS/DDoS	Komal et al. (2014) [83], Almori et al. (2012) [84] Porwal et al. (2014) [85]
	Sleep deprivation	Vimal et al. (2012) [86], Hasrouny et al. (2017) [87]
	Jamming attacks	Hasrouny et al. (2017) [87], Azer et al. (2014) [88]
	Jellyfish attack	Vimal et al. (2012) [86], Sakiz et al. (2017) [89]
	Intelligent cheater attack	Sakiz et al. (2017) [89]
	Blackhole attack	Kshirsagar & Patil (2013) [90]
	Grayhole attack	Sen et al. (2007) [91]
	Greedy behaviour attack	Mejri et al. (2014) [92]
	Spamming attack	Sumra et al. (2011) [76]

Continued on next page

Table 1 – *Continued from previous page*

Attacked Service	Type of Attack	Reference(s)
Authenticity	Sybil attack	John et al. (2015) [93], Doucear J.R.(2002) [94]
	Tunnelling attack	Sheikh et al. (2019) [95]
	GPS spoofing	Gamal et al. (2020) [72]
	Free-riding attack	Shilpa et al. (2015) [96]
	Certificate/ key replication attack	Junaid et al. (2018) [79]
Non-repudiation	Repudiation attack	Li et al. (2014) [97]

2.7.2 Attacks based on attacker-type:

This section discusses different types of attackers in VANETs, and how they launch attacks using various methods [42], [98]. Different categories of attackers include active and passive attackers, internal and external attackers, rational and malicious attackers, timing attackers, communication attackers, and area attackers. The classification of attackers based on their nature and activities is essential in developing appropriate countermeasures to ensure the security of vehicular networks. Identifying and understanding the various types of attackers and their techniques is crucial for devising effective security measures for vehicular networks. Attackers compromise the safety and privacy of the vehicles and their passengers, making it necessary to have adequate security measures in place. The different categories of attackers and their activities provide insights into the potential security risks and this knowledge helps in developing robust security mechanisms to safeguard vehicular networks. Moreover, vehicular networks are becoming more complex with the increasing number of connected vehicles and the growing use of autonomous driving technologies. These developments also bring about new security challenges that need to be addressed. The presence of different types of attackers with varying skills and resources can make it challenging to ensure the security of vehicular networks. Therefore, a comprehensive and proactive approach is required to address these challenges and maintain the safety and security of vehicular networks.

Attackers might be classified as either external or internal. Internal attackers exist as an intrinsic part of the VANET and are fully aware of it. On the other hand, external attackers remain outside the VANET and are unaware of its architecture.

Another classification of attackers is based on their intentions. Malicious and rational attackers are two types of attackers. Rational attackers launch attacks for personal gains, such as money or revenge. However, malicious attackers carry out assaults without financial gain.

An attacker that alters the time for conversation or introduces unwarranted lags in communication is regarded as a timing attacker [99]. A communication attacker chooses the network, such as V2V or V2I, as their goal and executes an assault on it.

An area attacker conducts an assault against a particular vehicle, group of vehicles, or location. It is crucial to comprehend these different types of attackers to create efficient security measures to defend vehicle networks from these dangers.

2.7.3 Attacks on VANET layers:

This section examines vehicular networks and their layers, analogous to the OSI model [100]. Each layer has predefined tasks, and different protocols are employed at different layers. This model does not include the Session and DLL, whereas the MAC and LLC layers are formed for various jobs. The following sections define each of these layers and discuss potential possible attacks.

Application layer: The Application layer is responsible for receiving user input and forwarding it to other layers [66]. This layer also accepts output from different layers and presents it to users. Attack possible against VANET application layer includes the following:

Message Tampering: Message tampering attacks focus on the integrity of data at the application layer. Attackers manipulate or modify messages exchanged between vehicles, potentially causing incorrect decisions or actions based on tampered information.

Impersonation Attack: Impersonation attacks involve malicious entities pretending to be legitimate vehicles or users within VANET applications. This attack can lead to unauthorized access and deceptive interactions.

Repudiation Attack: Repudiation attacks aim to undermine non-repudiation mechanisms at the application layer. Attackers may deny their involvement in certain actions or transactions, causing disputes and challenges in establishing trust.

Replay Attack: Replay attacks involve the unauthorized re-transmission of previously captured data. In VANET applications, this can result in the replay of outdated or incorrect information, impacting the reliability of communication.

Illusion Attacks: Illusion attacks create deceptive signals or data to mislead VANET applications, potentially causing vehicles to make erroneous decisions based on false information.

False Position Attacks: False position attacks involve providing incorrect location information within VANET applications, which can lead to safety hazards or misrouting.

Sybil Attack: Sybil attacks introduce multiple fake identities into the VANET application layer, disrupting trust and authenticity by creating deceptive entities.

Transport layer: The Transport layer ensures process-to-process delivery of messages and guarantees that they are sent in the proper order without any modifications [102]. It transports TCP and UDP packets from source to destination and keeps track of traffic movement and congestion on the network.

Replay Attack: Replay attacks, when targeted at the transport layer, involve the unauthorized retransmission of data packets. This can lead to issues such as repeated actions or message duplication, affecting the reliability of communication.

Tunnel Attacks: Tunnel attacks divert traffic through unauthorized channels or tunnels, potentially compromising data confidentiality and authenticity within the transport layer.

Man-in-the-Middle Attack: Man-in-the-middle attacks occur when attackers intercept and possibly modify data between communicating parties, undermining the confidentiality and integrity of data within the transport layer.

Message Tampering: Similar to the application layer, message tampering attacks at the transport layer focus on manipulating or altering data packets, impacting the integrity of communication.

Session Hijacking Attack: Session hijacking attacks involve unauthorized access to and control of ongoing communication sessions, leading to data exposure or manipulation.

Sybil Attack: Sybil attacks within the transport layer disrupt trust and authenticity by introducing multiple deceptive identities, potentially affecting the security of communication.

Network layer: The Network layer is responsible for data packet propagation from one node to another in vehicular networks [103]. In VANETs, security concerns differ from those in other networks because of many reasons. Consequently, the assaults made against VANET are unique.

Location Disclosure: Location disclosure attacks at the network layer involve revealing the physical locations of vehicles, which can compromise privacy and potentially enable tracking or targeting.

Packet Dropping: Packet-dropping attacks disrupt communication by intentionally discarding data packets, causing data loss and affecting the reliability of data transmission.

Flooding Attack: Flooding attacks flood the network with excessive data packets, causing congestion and potentially disrupting network services.

Replay Attack: Similar to other layers, replay attacks involve the unauthorized retransmission of data packets within the network layer, affecting data integrity and reliability.

Message Tampering: Message tampering attacks focus on manipulating or altering data packets in transit within the network layer, impacting data integrity.

Sybil Attack: Sybil attacks within the network layer introduce multiple deceptive identities, undermining trust and potentially disrupting network operations.

Wormhole Attack: Wormhole attacks involve the creation of a covert, high-speed link between distant parts of the network, facilitating unauthorized communication and potentially compromising data confidentiality and integrity.

Blackhole Attack: Blackhole attacks involve intercepting and discarding data packets, making them inaccessible to other vehicles or nodes within the network, affecting data availability and integrity.

Tunnel Attack: Tunnel attacks divert network traffic through unauthorized routes or tunnels, potentially compromising data confidentiality and authenticity.

LLC and MAC layer: Multiple approaches manage traffic jams at the LLC and MAC layers. Congestion management strategies might be reactive, proactive, as well as blended. [104].

DoS and DDoS Attacks: Similar to other layers, DoS and DDoS attacks at the LLC/MAC layer disrupt communication and network services by overwhelming network resources.

Illusion Attacks: Illusion attacks at the LLC/MAC layer create deceptive signals or data, potentially misleading network protocols and impacting data reliability.

Routing Attack: Routing attacks involve the manipulation of routing protocols, leading to incorrect routing decisions and potential data misdirection.

Signal Jamming Attack: Signal jamming attacks disrupt wireless communication by emitting interference signals, potentially leading to packet loss and communication breakdown.

Replay Attack: Replay attacks within the LLC/MAC layer involve the unauthorized retransmission of data frames, affecting data integrity and potentially causing data duplication.

Impersonation Attacks: Impersonation attacks target the authenticity of communication by involving malicious entities posing as legitimate vehicles or nodes within the LLC/MAC layer.

Message Tampering: Message tampering attacks within this layer focuses on the manipulation or alteration of data frames in transit, impacting data integrity.

Sybil Attack: Sybil attacks within the LLC/MAC layer introduce multiple deceptive identities, disrupting trust and potentially affecting network performance.

Collision Attack: Collision attacks occur when multiple vehicles or nodes transmit data simultaneously on the same channel, leading to data collision and potential data loss or corruption within the LLC/MAC layer.

Physical layer: The Physical layer uses 802.11p OFDM in the dedicated short-range communication [104]. It operates in the frequency spectrum of the 5.9 GHz band (5.885–5.905) with a 10 MHz wide channel in the WAVE. The data rate in this type of communication is typically 3 Mbps, with a default data rate of 6 Mbps. Researchers use various aspects of the physical layer, including transmission power control, using multiple (or single) antennas, channel estimation, and channel selection.

GPS Spoofing Attack: GPS spoofing attacks manipulate GPS signals to deceive vehicles about their physical location, compromising the accuracy of location-based services and navigation within the physical layer.

Jamming Attack: Jamming attacks emit radio interference, disrupting wireless communication channels and potentially leading to packet loss and communication breakdown at the physical layer.

Message Altering Attack: Message-altering attacks target the integrity of data frames at the physical layer by manipulating or altering data in transit. A message-altering attack or a message tampering attack, is a type of cybersecurity attack where an unauthorized party intercepts a message in transit and modifies its content before allowing it to reach its intended recipient.

Passive Eavesdropping: Passive eavesdropping involves unauthorized entities listening in on wireless communication, potentially leading to a breach of confidentiality and privacy within the physical layer.

Routing Attack: Routing attacks are malicious activities aimed at disrupting or manipulating the routing of data packets in computer networks. These attacks can involve altering routing tables, injecting false routing information, or diverting network traffic leading to network disruptions.

Table 2 demonstrates various attacks possible on VANET layers.

Table 2: Attack classification based on VANET layers

Attacked Layer	**Type of Attack**	**Reference(s)**
Application layer	DoS & DDoS	Komal et al. (2014) [83], Almori et al. (2012) [84], Porwal et al. (2014) [85],
	Message tampering	Singh & Sharma (2019) [81]
	Impersonation attack	Tyagi et al. (2014) [105]
	Repudiation attack	Li et al. (2014) [74]
	Replay attack	Junaid et al. (2018) [79], Malik et al. (2019) [80]
	Illusion attacks	Lo & Tsai (2007) [82]
	False position attacks	Gamal et al. (2020) [72]
	Sybil attack	John et al. (2015) [93], Douceur J.R.(2002) [94]
Transport layer	DoS and DDoS attack	Komal et al. (2014) [83], Almori et al. (2012) [84], Porwal et al. (2014) [85],
	Replay attack	Junaid et al. (2018) [79], Malik et al. (2019) [80]
	Tunnel attacks	Sheikh et al. (2019) [95]
	Man in the middle attack	Ahmad et al. (2018) [73], Li et al. (2012) [74]
	Message tampering	Singh & Sharma (2019) [81]
	Session hijacking attack	Hasrouny et al. (2017) [87]

Table 2: Attack classification based on VANET layers

Attacked Service	Type of Attack	Reference(s)
	Sybil attack	John et al. (2015) [93], Douceur J.R.(2002) [94]
Network layer	Location disclosure	Mansour et al. (2018) [106]
	Packet dropping	Mansour et al. (2018) [106]
	Flooding attack	Vimal et al. (2012) [86]
	Replay attack	Junaid et al. (2018) [79], Malik et al. (2019) [80]
	DoS and DDoS attack	Komal et al. (2014) [83], Alomari et al. (2012) [84] Porwal et al. (2014) [85]
	Message tampering	Singh & Sharma (2019) [81]
	Sybil attack	John et al. (2015) [93], Douceur J.R.(2002) [94]]
	Wormhole	Sen et al. (2007) [91]
	Blackhole attack	Kshirsagar & Patil (2013) [90]
	Tunnel attack	Sheikh et al. (2019) [95]
LLC/MAC layer	DoS and DDoS attack	Komal et al. (2014) [83], Almori et al. (2012) [84], Porwal et al. (2014) [85]
	Illusion attacks	Lo & Tsai (2007) [82]
	Routing attack	Kong et al. (2003) [107]
	Signal jamming attack	Karagiannis & Argyriou (2018) [108]
	Replay attack	Junaid et al. (2018) [79], Malik et al. (2019) [80]
	Impersonation attacks	Tyagi et al. (2014) [80]
	Message tampering	Singh & Sharma (2019) [81]
	Sybil attack	John et al. (2015) [93], Douceur J.R.(2002) [94]
	Collision attack	Tolba Amr (2018) [109], Mayank et al. (2016) [110]
Physical layer	DoS and DDoS attack	Komal et al. (2014) [83], Almori et al. (2012) [84], Porwal et al. (2014) [85]
	GPS spoofing attack	Gamal et al. (2020) [72]
	Jamming attack	Hasrouny et al. (2017) [87], Azer et al. (2014) [88]
	Message altering attack	Singh & Sharma (2019) [81]
	Passive eavesdropping	Choudhari et al. (2019) [77]
	Routing attack	Kong et al. (2003) [107]

2.7.4 Attacks on VANET components:

Attackers launch attacks on vehicular networks by targeting three categories of components: vehicles, information, and infrastructure.

Vehicles: Vehicles consisting of OBU and AU are the least secure units in VANETs and can be easily targeted by attackers. Social engineering attacks, sensor impersonation attacks, and malware integration attacks are among the types of attacks possible against these units. Attackers can also target information flowing through the network, which can be compromised by novel attacks such as eavesdropping, jamming, spoofing, and false position attacks. These attacks can affect both the safety and non-safety applications of the network.

Physical Damage to Vehicle: In this attack, malicious actors physically damage vehicles, potentially compromising their safety and functionality. This type of attack can result in accidents and pose a direct threat to the well-being of passengers.

Sensor Impersonation Attack: Attackers impersonate the sensors of a vehicle, providing false or misleading information to the vehicle's control systems. This can lead to incorrect decision-making by the vehicle and jeopardize safety.

Bogus Information Attack: Malicious entities inject false or bogus information into a vehicle's communication system, potentially causing confusion and incorrect actions by the vehicle.

Illegal Remote Firmware Attack: This attack targets a vehicle's firmware remotely, compromising its operational integrity. Attackers can gain unauthorized control over vehicle functions, posing significant safety and security risks.

Jamming Attack at Vehicle Level: Jamming attacks disrupt a vehicle's communication by emitting radio interference. This interference can disrupt critical vehicle-to-vehicle (V2V) or vehicle-to-infrastructure (V2I) communication, potentially leading to accidents or miscommunications.

Social Engineering Attack: Social engineering attacks manipulate individuals within the VANET ecosystem, such as drivers or maintenance personnel, to gain unauthorized access to vehicles or sensitive information.

Malware Integration: Malicious software (malware) is introduced into a vehicle's systems, allowing attackers to gain control, steal information, or interfere with vehicle operations.

Credential Revelation: Attackers reveal or steal the credentials (e.g., usernames and passwords) of vehicles, potentially gaining unauthorized access and control.

Attacks on Information: Attacks on Information within Vehicular Ad-hoc Networks (VANETs) encompass a range of malicious activities aimed at manipulating or compromising data exchanged between vehicles and infrastructure. These attacks include fake information injection, impersonation, false position reporting, and message tampering. Fake information attacks involve the insertion of deceptive data, potentially leading to incorrect decisions by vehicles. Other attacks include the following:

Fake Information Attack: Fake information attacks involve injecting false data into the VANET communication. This can mislead vehicles, leading to incorrect decisions and actions.

Impersonation Attack: Impersonation attacks involve malicious entities posing as legitimate vehicles or nodes within the VANET, potentially gaining unauthorized access and compromising trust.

False Position Attack: Attackers provide incorrect position information within the VANET, potentially leading to navigation errors or safety hazards.

Message Tampering: Message tampering attacks manipulate or alter data messages exchanged within the VANET, potentially compromising data integrity.

Eavesdropping: Eavesdropping attacks involve unauthorized entities listening in on VANET communication, potentially leading to breaches of confidentiality and privacy.

Man-in-the-Middle Attack: In a man-in-the-middle attack, attackers intercept and possibly modify data exchanged between vehicles, compromising the confidentiality and integrity of data.

Spoofing Attack: Spoofing attacks involve the deceptive manipulation of data or signals within the VANET, potentially causing vehicles to make incorrect decisions based on false information.

Jamming Attacks: Similar to vehicle-level jamming attacks, these attacks disrupt VANET communication by emitting radio interference, leading to communication breakdown and potential safety issues.

Infrastructure components: Infrastructure components, such as Roadside Units (RSUs), central registration agencies, charging spots for EVs, trusted authorities, and video cameras placed alongside the road or at other locations, such as parking areas, can also be targeted by attackers. Attackers can launch network attacks, DoS/DDoS attacks, Sybil attacks, and man-in-the-middle attacks against infrastructure components. Another component that attackers can target in vehicular networks is the Global Navigation Satellite System (GNSS) for location-based services. Attackers can interfere with GNSS signals to mislead vehicles and cause accidents or perform spoofing attacks to impersonate the position of legitimate vehicles. Attackers can also target vehicular networks' Communication Infrastructure (CI) components. The CI includes the communication equipment and protocols used to connect the various network components. Attackers can exploit vulnerabilities in communication protocols or launch attacks such as network scanning and sniffing to intercept and manipulate network traffic. In addition, the Cloud Computing component of vehicular networks is another potential target for attackers. Cloud computing provides computational and storage resources to vehicular networks. Attackers can exploit vulnerabilities in the cloud infrastructure or launch attacks such as data theft, data manipulation, or Denial-of-Service (DoS) attacks to disrupt the services provided by the cloud. The following Table 3 demonstrates various possible attacks on VANET components.

Table 3: Attack classification based on VANET components

Component	**Types of Attack**	**Reference(s)**
Vehicles	Physical damage to vehicle	Sumra et al. (2011) [76]
	Sensor impersonation attack	Rawat et al. (2012) [111]
	Bogus information attack	Singh & Sharma (2019) [81]
	Illegal remote firmware attack	Dennis & Larson (2009) [112]
	Jamming attack at vehicle level	Hasrouny et al. (2017) [87], Azer et al. (2014) [88]
	Social engineering attack	Sumra et al. (2011) [76]
	Malware integration	Hasrouny et al. (2017) [87]
	Credential revelation	Whyte et al. (2013) [113]
Information	Fake information attack	Singh & Sharma (2019) [81]
	Impersonation attack	Tyagi et al. (2014 [105])
	False position attack	Gamal et al. (2020) [72]
	Message tempering	Singh & Sharma (2019) [81]
	Eavesdropping	Choudhari et al. (2019) [77]

Continued on next page

Table 3 – *Continued from previous page*

Component	Type of Attack	Reference(s)
	Man in the middle attack	Ahmad et al. (2018) [73], Li et al. (2012) [74]
	Spoofing attack	Gamal et al. (2020) [72]
	Jamming attacks	Hasrouny et al. (2017) [87], Azer et al. (2014) [88]
Infrastructure	Man in the middle attack	Ahmad et al. (2018) [73], Li et al. (2012) [74]
	Eavesdropping	Choudhari et al. (2019) [77]
	GPS tracking attack	Singh & Sharma (2019) [81]
	Sybil attack	John et al. (2015) [93], Doucear J.R.(2002) [94]
	Network attacks	Sumra et al. (2011) [76]
	Bogus information	Singh & Sharma (2019) [81]
	DoS and DDoS attack	Komal et al. (2014) [83], Alomari et al. (2012) [84], Porwal et al. (2014) [85]
	Wormhole attack	Sen et al. (2007) [91]

2.7.5 Attacks on electric vehicles:

Electric vehicles and their related infrastructure are also vulnerable to various types of cyber-attacks, which can impact the safety and performance of the vehicles. One type of attack is a relay attack, in which the attacker intercepts and relays communication between the vehicle and its keyless entry system, allowing them to unlock and steal the vehicle. Another type of attack is the DoS attack, which can disable the charging infrastructure and prevent EV owners from charging their vehicles. Malware and ransomware attacks can also be launched against the charging infrastructure or the vehicle's onboard computer system, compromising the safety and security of the vehicle's occupants. To mitigate these risks, EV manufacturers and charging infrastructure providers implement various security measures, including encryption, authentication, and intrusion detection systems. Policymakers and regulators need to consider cybersecurity in their regulations and standards for electric vehicles and their infrastructure to ensure the safety and security of these vehicles in the rapidly evolving transportation landscape.

According to Wang [66], EVs communicate with smart grids, vehicles, and infrastructure, which makes them an essential component of VANETs. As the automobile industry shifts its focus towards EVs, their availability has increased globally, and their numbers are rising. EVs use intelligent charging systems to charge their batteries, where the EV and charging device share data connections with the charging operator. EVs use Grid-to-Vehicles (G2V) or Vehicle-to-Grids (V2G) systems for charging. In the G2V system, electricity moves in one direction from the charging point or grid to the vehicle. In the V2G system, electricity moves in both directions, allowing for the exchange of surplus electricity between the vehicle and the charging point. Saxena [115] explains that this type of system works by utilizing intelligent grids. Li [74] suggests a more advanced charging system in which EVs exchange electricity through vehicle-to-vehicle charging.

EVs connect to charging stations and billing/payment systems, also related to smart grids for communication. However, communication between these components is vulnerable to attacks such as eavesdropping, man-in-the-middle attacks, and repudiation attacks. Falk [117] identifies different attack-vulnerable EV assets, such as access control policies, time, configuration data, software, firmware and drivers, control commands, clock settings, meter data, tariff data, customer ID, and location data. These assets are vulnerable to attacks, including those affecting other vehicular networks. Although ISO/IEC standards work to secure charging systems and smart grids, they require further improvement since attackers possess various techniques to launch attacks against them. In addition to attacks on other vehicles, the infrastructure used for vehicle charging and electricity distribution systems is also vulnerable, necessitating additional security measures for EVs.

2.8 Conclusion

This chapter provides a comprehensive literature review of VANETs, a type of mobile ad-hoc network within the IoT. The text overviews VANETs, including their architecture, applications, and features. It also discusses the challenges faced by VANETs, such as mobility management, security, and scalability. The impact of security on the performance of VANETs is also examined, with a thorough analysis of security in VANETs, highlighting the various security requirements. The text provides a classification of different types of attacks based on the attacker, VANET layers, components, and electric vehicles (EVs). The text concludes by summarizing the insights from the literature. The future of vehicular networks, including VANETs, is described as having the potential to save the lives of millions of people worldwide by providing a safer and more efficient transportation system. However, ensuring security in VANETs is emphasized to ensure their successful deployment and continued growth.

Chapter 3

Security Solutions

In this chapter, we explore various security solutions aimed at mitigating DDoS attacks in VANET. This chapter examines security solutions in detail and discusses their strengths and limitations. By analyzing these security solutions, this thesis chapter aims to provide insights into the design and development of secure and trustworthy VANETs. The objective is to provide a comprehensive overview of the existing literature in this area and to evaluate the effectiveness of existing security solutions. We begin by reviewing the different attack detection techniques. These techniques include identity-based solutions, key-based solutions, and trust-based solutions, as well as machine learning (ML) and hybrid solutions. We evaluate the strengths and weaknesses of these techniques and discuss how they detect DDoS attacks in VANET. In addition to attack detection, we also study various DDoS prevention techniques. We evaluate the effectiveness of these techniques and analyze their impact on the performance of VANET. Finally, we discuss the challenges that need addressing for effective deployment of these security solutions in real-world scenarios.

3.1 Introduction

VANETs pose unique security challenges, as they operate in open and dynamic environments and are vulnerable to various attacks. Security threats compromise these networks' safety, privacy, trustworthiness, and reliability, leading to potential accidents, loss of life, and damage to critical infrastructure. Therefore, it is essential to implement adequate security solutions in VANETs to make sure the network is secure, reliable, and trustworthy. Furthermore, trust and reliability are crucial factors for the success of VANETs. Users must be confident that the information received from the network is accurate and timely and that the network operates smoothly without disruptions. However, security threats can undermine trust in the network, resulting in reduced user adoption and deployment. Similarly, network disruptions caused by security threats can lead to downtime, delays, and lost data, impacting the reliability of VANETs. The need for security solutions in VANETs arises because these networks rely heavily on exchanging sensitive information among vehicles and infrastructure components. For instance, safety applications such as collision avoidance and traffic signal preemption require real-time communication between vehicles and infrastructure. Any security breach in such applications can compromise the safety of passengers and other road users. Similarly, collecting and transmitting sensitive information in VANETs, such as location data, can lead to serious privacy violations if not adequately secured. Various security solutions have been proposed for VANETs to address these security challenges. We can broadly categorize these solutions into attack detection and prevention techniques. Attack detection solutions stop security threats and identify the attacker, whereas prevention solutions stop attacks from happening in the first place. It is crucial to use adequate security measures in vehicular networks for the following reasons:

1. Safety: Vehicular networks are critical for communication and coordination between vehicles, and essential for providing road safety. Security threats can disrupt these networks leading to accidents and loss of life.

2. Privacy: Vehicular networks collect and transmit sensitive information about vehicles, passengers, and the environment. Security threats can expose this information to malicious actors, resulting in severe privacy violations.

3. Trust: The trustworthiness of vehicular networks is essential for their widespread adoption and deployment. Security threats can erode trust in these networks, making them less attractive to users and hindering their growth.

4. Reliability: Vehicular networks must be highly reliable to guarantee safe and efficient vehicle communication. Security threats can disrupt the normal functioning of these networks, leading to downtime, delays, and lost data. Tanwar et al. classify security solutions for vehicular networks into two main categories: attack detection solutions and prevention solutions [118]. This chapter covers both categories of solutions.

3.2 Detection solutions

Attack detection refers to identifying and classifying malicious or anomalous activities that try to compromise the security of a system, network, or application. An attack detection system is necessary for any effective cybersecurity strategy, as it allows organizations to detect and respond to potential threats on time. Attack detection involves checking various security events and traffic patterns and analyzing them for signs of suspicious or malicious behavior. It also generates alerts for security administrators or automated systems when potential threats are detected. Attack detection includes identifying abnormal user activity, network traffic patterns, or system behaviors that may indicate an attack or security breach. Attack detection techniques include tools and technologies like IDS and Security Information and Event Management (SIEM) systems. These tools use machine learning algorithms and other advanced analytics techniques to identify and classify potential threats. These tools also incorporate automated response capabilities to mitigate the impact of attacks.

3.2.1 Identity based solutions

Identity-Based Cryptography (IBC) is a cryptographic approach Adi Shamir introduced in 1984. Boneh and Franklin made IBC practical when they implemented it in 2001 [119]. Unlike Public Key Infrastructure (PKI), which uses certificates to verify public keys, IBC solutions generate the public key using the identity information of an entity, such as name, IP address, or email address. In 2001, Boneh and Franklin introduced the IBC system using bilinear pairings on elliptic curves [119]. IBC categorizes the solutions into Identity-Based Encryption (IBE) and Identity-Based Signatures (IBS). IBC comprises IBE and IBS, which means IBC contains IBE and signatures. These solutions are computationally more efficient due to the absence of certificate verification.

Table 4: Identity based solutions

Reference	**Service**	**Attack type**	**Solution**	**Claims**
Zhang et al. (2002) [120]	Anonymity and privacy	Signature forgery	ID-based blind signature and ring signature from pairing	Can be used in electronic cash scheme or electronic voting scheme.
Choon & Cheon (2002) [121]	Confidentiality and Authenticity	Forgery	ID-based signature using gap Diffie Hellman (GDH) groups	Small signature size with more efficiency.
Chow et al. (2005) [122]	Confidentiality, Authenticity, non-repudiation	Message and identity attack	ID-based ring signature	Requires only two pairing computations and un forgeable
Gamage et al. (2006) [123]	Confidentiality, Authenticity	Forgery	ID based ring signature.	Less computational overheads
Kamat et al. (2006) [124]	authentication, confidentiality, non-repudiation and message integrity	Modification attack, man in the middle, Replay attack	ID-based security framework	No extra memory requirements for pseudonym
Jinyuan et al. (2010) [125]	Authentication, non-repudiation, integrity, confidentiality.	Forgery, man in the middle, Replay attack	Novel pseudonym-based scheme and threshold signature-based security system	Efficient in storage, computations and communication
Lim and Paterson (2010) [126]	Confidentiality, Authenticity	Impersonation attack, modification attack	Identity-based key infrastructure for the grid (IKIG)	Least costs of computation & communication with employability in smart grids.
He et al. (2015) [127]	Confidentiality/ Privacy	Impersonation attack, modification attack, man in the middle, Replay attack, stolen verifier table attack	ID based Conditional Privacy-Preserving Authentication (CPPA)	Low computation cost and communication cost

Table 4: Identity based solutions

Reference	Service	Attack type	Solution	Claims
Ali et al. (2019) [128]	Authentication	Forgery	Elliptic Curve Cryptography (ECC) and general one-way hash functions	Single signature cost = 0.4438 ms, Batch signature cost= 0.442 + 0.0018n ms
Limbasiya et al. (2019) [129]	Authentication and privacy	Impersonation attack, modification attack man in the middle, Replay attack, Session key enclosure	proficient message verification scheme	Computational cost =10Th + 3TECM, Time= 1.5970ms, Energy (mJ)= 17.3754
Al-shareeda et al. (2020) [130]	Privacy	Impersonation attack, modification attack man in the middle, Replay attack,	VANET based privacy-preserving communication scheme (VPPCS)	optimum values of MGS(ms), SV M(ms), BV MM(ms)

3.2.2 Key based solutions

VANETs rely on security solutions based on cryptographic keys to ensure the confidentiality, integrity, and authenticity of transmitted data within the network. These key-based solutions use cryptographic keys and hash functions to establish secure communication between nodes in the network, with public keys used for encryption and private keys for decryption. Key-based solutions guarantee the integrity of data transferred over the network and help prevent unauthorized access to critical information. Key-based solutions, such as PKI, symmetric, and asymmetric keys, are crucial for maintaining the security of VANETs and provide a secure mechanism for exchanging data between nodes while protecting against various types of attacks. It is worth noting that key-based solutions differ from ID-based ones in that they use certificates to establish the authenticity of the communicating nodes. These solutions are widely used to secure communication in VANETs and are essential for ensuring the reliability and security of these networks [131].

Table 5: Key based solutions

Reference	Service	Attack type	Solution	Claims
Sanzgiri et al. (2002) [132]	Authentication, non-repudiation	Replay attack Impersonation Eavesdropping	ARAN Protocol	Efficient and simple protocol
Hu et al. (2002) [133]	Availability, , non-repudiation	DoS Routing attack Replay attack	Ariadne	41.7% lower packet overhead
Hu and Johnson (2003) [134]	Authentication, availability	DoS Routing attack Impersonation	SEAD Protocol	Packet delivery ratio is almost 95%
Cencioni et al. (2008) [75]	Confidentiality	Traffic Analysis Attack	VIPER: a Vehicle-to-Infrastructure communication Privacy Enforcement pRotocol (Cryptography)	Efficient, low computations, less time delays and less dummy messages sent,
Li et al. (2012) [74]	Confidentiality	Man in the Middle	Mobile Payment Protocol (Cryptography)	Reduced communication and computation cost

3.2.3 Trust based solutions

In VANET, trust-based solutions manage security threats and prevent unauthorized access to sensitive information transmitted over the network. These solutions utilize a trust model that establishes trust relationships between nodes in the network. We do this to determine which nodes are trustworthy for exchanging information and which nodes are not. We define the trust relationship based on predefined criteria like node behavior, reputation, and history. Trust-based solutions offer an alternative to key-based ones, where public and private keys secure communication. Instead, they rely on trust relationships between nodes to ensure the confidentiality, integrity, and authenticity of information transmitted over the network. Reputation-based systems, where nodes are assigned a reputation score based on their past behavior, and trust management systems, where nodes negotiate trust relationships with other nodes, are examples of trust-based solutions in VANET. VANETs use various trust models in VANET, including data-oriented and entity-oriented models. Data-oriented models focus on the data used during communication, while entity-oriented models emphasize the reliability of the drivers of vehicular entities.

Table 6: Trust based solutions

Reference	Service	Attack type	Solution	Claims
Poongodi et al. (2019) [135]	Availability	DoS/DDoS	A trust-based system where history and profiles are observed	Det. rate =95.8%, Avg. latency = 30s, P Del.R = 86%
Nandy et al. (2020) [136]	Availability	DoS/DDoS	T-BICDS, trust-based collaborative intrusion detection system	Detection rate and accuracy not defined

3.2.4 Machine Learning solutions

Machine learning solutions use artificial intelligence algorithms to detect and prevent security threats in VANETs. ML solutions automatically learn from network-generated data, identify patterns and anomalies, and respond to security threats in real time. ML solutions for VANETs detect various types of attacks, such as DDoS attacks, message tampering, and eavesdropping. These solutions identify malicious nodes in the network using different ML algorithms, such as decision trees, artificial neural networks, and support vector machines. These algorithms analyze large amounts of data and decide how to respond to security threats. Examples of these solutions include IDS, which uses ML algorithms to detect malicious behavior in the network, and anomaly detection systems, which use ML algorithms to identify and respond to unusual patterns in network traffic. These solutions are becoming popular in VANETs because they quickly detect and respond to security threats in real time, even as the network evolves and changes over time.

Table 7: Machine learning solutions

Reference	Service	Attack type	Solution	Claims
Grover et al. (2011) [137]	Availability	DoS/DDoS	Random forest, naïve-bayes, IBK, J-48, ada-boost1	In Binary Classifier = 92% In multi classifier = 93%

Table 7: Machine learning solutions

Reference	**Service**	**Attack type**	**Solution**	**Claims**
Li et al. (2015) [138]	Availability	DoS/DDoS	Support vector machine	Better performance in terms of accuracy, precision, and recall
Ghaleb et al. (2017) [139]	Availability	DoS/DDoS	Artificial neural network (ANN)	Accuracy= 99.74%, Detection Rate = 99%
Kim et al. (2017) [140]	Availability	DoS/DDoS	Multi-class support vector machine	Accuracy ¿ 85%, Precision = 90%, Recall =85%
Yu et al. (2018) [141]	Availability	DoS/DDoS	Support vector machine	Accuracy more than 97%
Karagiannis & Argyriou (2018) [108]	Availability	DoS/DDoS	Unsupervised learning with clustering (k-means)	Not Available
Liang et al. (2011) (2018) [142]	Availability	DoS/DDoS	I-GHSOM	Accuracy= 99.69%
Kosmanos et al. (2019) [143]	Availability	DoS/DDoS	k-NN and the Random Forest algorithm	Accuracy= 91%
Kaur et al. (2019) [144]	Availability	DoS/DDoS	Adaptive neuro-fuzzy system	Good efficiency in terms of packet loss rate, delay and throughput
Aloqaily et al. (2019) [145]	Availability	DoS/DDoS	Decision Tree	Accuracy =99.43% and detection rate = 99.92%
Kolandaisamy et al. (2019) [146]	Availability	DoS/DDoS	EBACA scheme based on ant colony optimization and integrated Markov-chain	Attack mitigation 78% more than other solutions.
Zeng et al. (2019) [147]	Availability	DoS/DDoS	DeepVCM using CNN and LSTM	Good performance in accuracy, precision and recall
Manimaran et al. (2020) [148]	Availability	DoS/DDoS	Heuristic-based adaptive IDS	Not Available

Table 7: Machine learning solutions

Reference	Service	Attack type	Solution	Claims
Shahverdy et al. (2020) [149]	Availability	DoS/DDoS	CNN	Accuracy= 99. 95%
Schmidt et al. (2020) [150]	Availability	DoS/DDoS	Knot flow classification with spline implementation	Accuracy = ¿73% with all classifiers
Adhikary et al. (2020) [151]	Availability	DoS/DDoS	AnovaDot and RBFDot in SVM	Good performance in accuracy, Gini, KS, MER and H parameters
Liu et al. (2020) [152]	Availability	DoS/DDoS	Naive-Bayes classifier	Not provided

Several machine learning-based solutions are available for misbehavior detection in vehicular networks. Grover et al. (2011) use NCTUns-5.0 and WEKA for DDoS detection and show the efficiency of IDS using random forest, Naive-Bayes, IBK, J-48, and Adaboost1 [184]. Li et al. (2015) propose a context-aware ML-based solution called SVM-CASE that uses SVMs to detect security threats in VANETs. It considers the context of the network, such as the location, velocity, and acceleration of vehicles, as well as the traffic conditions, to make more accurate security decisions [185]. Ghaleb et al. (2017) suggest a model that uses ANN to detect misbehavior in VANETs such as the black hole, grey hole, wormhole, and Sybil attacks [186]. Kim et al. (2017) propose a collaborative security attack detection mechanism that uses Multi-class SVM in MATLAB and KDD CUP-1999 dataset to predict the attacks [187]. Yu et al. propose an SDN-based system for detecting DDoS attacks in a vehicular environment, which uses differences in traffic flow and position as parameters for attack detection and applies SVM [188].

D. Karagiannis and A. Argyriou (2018) propose a new IDS against spoofing attacks in connected EVs using unsupervised learning with clustering (k-means) and R-Studio tool, which uses Relative Speed Variations (RSV) as the detection parameter [189]. Similarly, Liang et al. (2018) suggest an Improved Growing Hierarchical Self-Organizing Map (I-GHSOM) that uses differences in traffic flow and position as parameters for attack detection, applies the Support Vector Machine (SVM) algorithm and uses the Mininet network simulator tool [190]. Kosmanos et al. (2019) propose a new solution to handle spoofing attacks in electric vehicular networks with an ML approach. It employs a clustering algorithm to group vehicles according to their charging patterns and uses a decision tree to classify them as usual or abnormal [191]. Kaur et al. (2019) provide an enhanced approach of an adaptive neuro-fuzzy system for attack detection in vehicular networks, which uses an ANFIS trained on a normal and attack traffic dataset using a combination of related features [192]. Aloqaily et al. (2019) propose an intrusion detection system for connected vehicles in smart cities using the decision tree algorithm and the NSL-KDD dataset, implemented in MATLAB [193]. Kolandaisamy et al. (2019) introduce a solution with an integrated Markov-chain and ant-colony optimization scheme using NS2 and Exploratory Based Ant Colony Approach (EBACA)[194]. Manimaran et al. (2020) propose NDNIDS, an intrusion detection system for NDN-based VANETs, using heuristic-based adaptive IDS and generating a dataset, and also uses multiple sensor values and the driver's heartbeat rate as a detection feature[195]. Schmidt et al. (2020) propose spline-based IDS, which works with knot flow classification [196]. Adhikary et al. (2020) present a hybrid algorithm that detects DDoS attacks in VANETs using AnovaDot and RBFDot in SVM for classification [197]. The algorithm employs a dataset synthesized by inducing jitter, delay, packet drop, throughput, and collision for attack detection. In a recent study, Kadam et al. (2021) propose a Hybrid KSVM scheme for intrusion detection that utilizes various parameters such as protocol, source and destination IP, and port no.[198]. Türkoğlu et al. (2022) introduce the SD-VANET architecture for intrusion detection in vehicular networks[199]. The architecture utilizes a feature selection algorithm based on Minimum Redundancy Maximum Relevance (MRMR) for classification.

3.2.5 Hybrid solutions

Hybrid solutions for securing VANETs refer to security measures that employ multiple security techniques to provide a more effective defense against security threats. Such solutions typically integrate traditional security techniques, such as encryption and authentication, with modern security techniques, such as machine learning algorithms, to create a more robust security solution. An example of a hybrid solution for VANET attacks might involve combining identity-based cryptography with machine learning algorithms to enhance security. In this scenario, identity-based cryptography would verify the authenticity of public keys used in communication. At the same time, the machine learning algorithms would detect and respond to security threats in real-time. Hybrid solutions address specific security challenges, such as DDoS attacks, message tampering, and eavesdropping. We customize them to meet the unique requirements of different VANET deployments. By integrating multiple security techniques, hybrid solutions offer a more comprehensive defense against security threats in vehicular networks and are a popular choice for organizations seeking to secure their VANETs.

Table 8: Hybrid solutions

Reference	**Service**	**Attack type**	**Solution**	**Claims**
Lo & Tsai (2007) [82]	Integrity	Illusion Attack	Plausibility validation network	Performance not quantified
Mejri et al. (2014) [92]	Availability	Greedy behavior attack	Entropy-based solution for DDoS detection	Detection and accuracy are not defined.
Malhi et al. (2016) [78]	Integrity	Masquerade	Secure genetic-based framework	Avg. accuracy rate of 3 scenarios is 86.54% with efficiency in packet drop ratio.
Lahrouni et al. (2017) [153]	Availability	DDoS attack	A mathematical model	Attack vehicle accuracy= 100% in all cases, Non attack vehicle accuracy in RMS= 87.96%, in MAV= 65.67% and in MSE = 0%
Malik et al. (2019) [80]	Integrity	Replay Attack	A non-learning based method	Not available
Li et al. (2020) [154]	Availability	DoS/DDoS	Heterogeneous communication framework with the hybrid information exchange	Detection rate and accuracy not defined

3.2.6 Solutions for EV infrastructure

EVs are not immune to cyber-attacks and attackers can target these EVs like any other vehicle. However, due to the lack of standardized communication procedures, EVs are at a higher risk of being compromised. The charging infrastructure for these vehicles is also susceptible to various security threats. Therefore, it is essential to implement security measures and protocols to safeguard EVs from potential attacks.

Table 9: Solutions for EV infrastructure

Reference	Service	Attack type	Solution	Claims
Wan et al. (2016) [155]	Authentication, privacy	Eavesdropping and active adversaries.	Privacy-preserving scheme PRAC, Privacy via Randomized Anonymous Credentials	Exact claims not available
Liu et al. (2018) [156]	Authentication, non-repudiation	Tampering attack	Blockchain-inspired data coins and energy coins	Exact claims not available
Kim et al. (2019) [157]	authentication	replay and man-in-the-middle	Blockchain-based charging systems	Communication cost = 1824 bits, computation cost = 1.1709 ms and 0.6104 ms
Marzougui et al. [158]	Energy management	Energy distribution	fuzzy logic control, a flatness control and rule-based algorithm	Efficient in performance
Kumar et al. (2020) [65]	Confidentiality, Authentication, non-repudiation	authentication attacks	Use of lattice-based cryptographic hash SWIFFT	avoids heavy computation and reduces communication cost up to 83%
Kavousi et al. (2020) [159]	Availability, authentication	Message flooding	Wavelet decomposition method and modified support vector machine	The high HR= 97.6 % and CR= 95.14 % as well as the low MR=2.4% and FR=4.86%

3.3 Prevention solutions

Preventive measures in VANETs aim to prevent security threats from occurring. We can accomplish this through various security technologies such as encryption, authentication, and firewalls. Additionally, preventive solutions in VANETs include utilizing advanced technologies like machine learning algorithms to detect and respond to security threats in real-time.

3.3.1 Conventional solutions

Conventional solutions for preventing attacks encompass a wide range of techniques utilized not only in computer networks but also in other types of networks, such as MANET and VANET. These solutions are typically based on established security practices, such as encryption, authentication, access control, and intrusion detection systems. They are often combined to provide a layered defense against security threats. While these conventional solutions can effectively mitigate attacks, they may only sometimes be sufficient to address the unique security challenges presented by emerging technologies and new network architectures. As such, there is a growing need for more advanced and adaptive security solutions that can keep pace with the evolving threat landscape.

Table 10: Conventional attack prevention techniques

Main Technique	**Solution**	**Author(s)**	**Functionality or Working**
Egress Filtering	Default Allow and Default Deny Policy	Mahajan & Sachdeva (2013) [209]	All outward moving packets are checked for any suspicious pattern for filtering
Ingress Filtering	Rule-Based ingress Filtering	Ferguson et al. (1998) [210]	Packets are not allowed to enter the network which does not follow the rules of the filter.
	Route Based Distributed Packet Filtering (DPF)	Park and Lee (2001) [211]	Proactive technique used for filtering based on the route used for traveling by the packet.
	History Based IP Filtering (HIF)	Peng et al. (2003) [212]	Filters packet after analyzing packet history record maintained in special databases.

Table 10: Conventional attack prevention techniques

Main Technique	Solution	Author(s)	Functionality or Working
	Hop Count Filter scheme (HCF)	Jin et al. (2003) [213]	Packets are filtered based on number of hops (nodes) traversed by the packet computed from TTL.
	Secure Overlay Services	Keromytis et al. (2004) [214]	Nodes are assigned special roles and tasks. Authentic packets are forwarded using hash functions.
Honeypot based Techniques	Simple Honeypots	Weiler N. (2002) [215]	Attackers are attracted by providing loopholes in fake security systems
	Interaction based honeypots	Daimi, K. (2018) [218], Kim et al. (2011) [219], NG et al. (2018),[220]	Interaction specifies how much activities are allowed by honeypot.
	Honeytokens	L. Spitzner, (2003) [221]	An entity or a keyword which attracts attacker.
	Honeynets	L. Spitzner, (2003) [222]	It is collection of two or more honeypots.
	Honeyfarms	Jiang et al. (2006) [223]	It is centralized group of honeypots.
Load Balancer	Benefit-based Load Balancing (BLB)	Anh et al. (2008) [225]	Traffic rate for forwarding further and for processing is controlled as per requirement.
Configuration Based Techniques	Disabling Unused Service	Fortunati et al. (2006) [227]	All those services which are not required should be disabled.
	Software Updates & Security patches	Fortunati et al. (2006) [227]	Software should be updated and patches should be installed immediately when available.
	Account privileges and authentication	Fortunati et al. (2006) [227]	User accounts should be maintained and permissions should be assigned.
	Changing IP addresses	Fortunati et al. (2006) [227]	IP addresses should be updated at regular intervals.
	Disabling IP Broadcasting	Fortunati et al. (2006) [227]	IP broadcasts should not be allowed.
Current Trends	DL-IDPS	Lee et al. (2020) [228]	Packet length is used as the main parameter for attack detection and prevention.
	3 tier IDPS	Ali et al. (2020) [229]	Software should be updated and patches should be installed immediately when available.

Table 10: Conventional attack prevention techniques

Main Technique	Solution	Author(s)	Functionality or Working
	Blockchain	Rohit et al. (2019) [230], Jamder et al. (2019) [231]	Blockchain-based solutions for IoT devices are proposed
	Umbrella	Liu et al. (2019) [232]	This solution works with the help of ISPs.
	reCAPTCHA controller	Poongodi et al. (2019) [233]	It works with the help of information based metrics.
	Statistics based solution	Ahuja and Singhal (2018) [237]	Statistical solutions are used for DDoS.
	MAEC-X architecture	Dao et al. (2018) [238]	Special MAEC-X architecture is proposed.
	D-PIDs	Sreerekha et al. (2018) [239]	It works with the help of information based metrics.

3.3.2 Logit model-based solutions

In this section, our focus is on solutions based on logistic regression and probit regression. These solutions are particularly advantageous in cybersecurity as they leverage probability distributions to prevent a range of security attacks effectively. Moreover, a related study proposed an early warning mechanism for estimating the likelihood of bank failure, which utilized detailed logit and discriminant analysis techniques [240].

Table 11: Logit and Probit Model-based prevention techniques

Author(s)	Strategy Used	Advantage	Parameters	Results
Martin, d.(1977) [240]	logit & discriminant analysis	Fund supply differs as per premium.	asset risk, liquidity, capital adequacy,	log odds ratios, marginal effect coefficients
Green et al. (2007) [241]	Binary grouped logit regression	data analysis of intrusion prevention systems to see hackers' behavior	protocol, the origin of the event, service accessed, and method of access.	false and true negative alerts report from the model.
Mukkhopadhyay et al. (2009) [242] [244]	Multinomial logit & probit regression	Propose e-risk insurance	time	quantify expected risk, attack trend projection

Table 11: Logit and Probit Model based prevention techniques

Author(s)	Strategy Used	Advantage	Parameters	Results
Jinkun et al.(2016) [243].	Probit regression	provided novel way to construct algorithm of VL model	time	predictions and validity for accumulative vulnerability loss
Prakash et al.(2020) [245]	i) support vector ii) random forest iii) logistic which includes newton-raphson,	evaluation through accuracy scores.	flow duration, forward packets, backward packets, total length of the packet	confusion matrix (performance analysis), precision results
Sharma et al. (2020) [246]	CRAM-D model-based logit-probit	targeted risk mitigation through heat matrix (probabilities*loss)	Bits per second (in gb) & duration (hrs)	generalized probabilities, expected losses, risk-severity matrix.

3.4 Conclusion

In conclusion, the security of vehicular networks is a critical issue that requires proper attention to ensure these networks' safe and reliable operation. This chapter has reviewed various security solutions to mitigate DDoS attacks in VANETs, including identity-based, key-based, trust-based, ML, and hybrid solutions. We can make decisions about which solutions are best suited to our needs. We can develop effective strategies for implementing these solutions in our networks by understanding the strengths and weaknesses of these solutions. It is important to note that the security of vehicular networks is an ongoing challenge and requires continuous monitoring and improvement. All organizations must adapt their security strategies to address these changing security needs as new threats emerge, and technology advances. Ultimately, securing vehicular networks aims to ensure the privacy, safety, and reliability of these networks and protect the drivers, passengers, and other users who rely on them for their daily needs.

Chapter 4

Research Methodology

Research methodology is a scientific approach to conducting research that includes selecting tools and strategies for gathering, analyzing, and interpreting data. Any research study's methodology is an essential component because it impacts the validity and reliability of the results. This chapter describes the research methods of DDoS detection and prevention in VANETs. It thoroughly explains the research design, data collection procedures, data analysis strategies, and ethical issues taken into account to maintain the study's validity. The research methodology chapter is vital since it directs the research process and guarantees that the results acquired are correct and genuine.

4.1 Introduction

This chapter starts by reviewing the research gap in the area of DDoS detection and prevention in Vehicular Ad-hoc Networks (VANETs), which serves as the basis for this study. The chapter then goes over the specific objectives of the research, which include examining different DDoS attack methods, assessing the efficiency of current DDoS detection and prevention methods in VANETs, and suggesting a novel mechanism suited to the unique features of VANETs. The chapter goes on to explore the necessity to solve the intricate security concerns connected with VANETs and their growing use in intelligent transportation systems, which is what motivated this study. The chapter concludes by thoroughly explaining the research methodology used in this study.

4.2 Research gap

Although various solutions can handle DoS/DDoS attacks, mathematical, statistical, knowledge-based, and soft computing-based solutions have been proposed. But still, these solutions need more accuracy and have detection delays. Because VANET vehicles have OBUs with very slow processors and low storage capacity, it is challenging for the vehicles to do processing to detect attacks themselves. Attackers, on the other hand, launch attacks against complete VANET with a variety of new tools and techniques.

- **To make the cities smart, VANETs and their related security challenges must be considered by all the countries for future Smart Cities. So this area has lots of potential for research.**

  Elaboration: This gap suggests that there is a need for research that specifically addresses the role of VANETs in smart city infrastructure. It implies that existing studies may not have fully explored how VANETs can contribute to the development of smart cities. In the context of our thesis objectives, we focus on assessing the current state of VANET security within the smart city context, including the detection and prevention of DDoS attacks.

- **In many network security solutions, the availability of information is an issue with the minimum importance. However, increasing attacks like DoS/DDoS on availability have proved that this factor is equally important to other security factors. So Dos/DDoS attacks need proper attention.**

  Elaboration: This gap highlights the significance of availability as a critical security factor, especially in the face of the rising threat of DDoS attacks. It suggests that existing security solutions might not adequately prioritize availability concerns. In our thesis, we explore the implications of DDoS attacks on VANET availability, assess the current state of DDoS prevention techniques, and propose frameworks to enhance availability protection within VANETs.

- **Attack prevention techniques are essential, which may save computational overheads of the detection and mitigation process. Based on the attack data, we prevent future attacks. For this, we require an optimal solution to have the ability to classify traffic based on previous results and identify attack packets.**

  Elaboration: This gap emphasizes the importance of proactive attack prevention techniques in reducing the computational overhead associated with detection and mitigation. It implies that current solutions might be more focused on reactive measures. In our thesis, we delve into the development of an optimal attack prevention framework that leverages machine learning (ML) and previous attack data to classify traffic and identify potential attack packets, aligning with your second and third objectives.

- **VANET solutions should be general, reliable, and universally applicable. So there is a need to formulate a solution that is feasible practically and economically.**

  Elaboration: This gap highlights the requirement for VANET security solutions to be practical, reliable, and economically feasible. It suggests that existing solutions may not fully meet these criteria. In our thesis, we investigate the feasibility and cost-effectiveness of proposed security mechanisms, ensuring that our ML-based detection and prevention frameworks are practical for real-world deployment.

- **Among available solutions area of machine and deep learning is comparatively new and still growing as new algorithms and tools are being developed. Existing ML techniques are not much used for attack prevention and detection in VANETs, which are very beneficial for handling DoS/DDoS attacks on classical networks. Checking the suitability of these existing and new ML technologies is still an untouched area.**

  Elaboration: This gap underscores the underutilization of ML techniques in VANET security, particularly for addressing DoS/DDoS attacks. It suggests that there is untapped potential in leveraging ML and deep learning algorithms to improve security in VANETs. In our thesis, we explore the suitability and effectiveness of existing and emerging ML technologies for DDoS detection and prevention in vehicular networks.

- **All security solutions must have accuracy in attack detection, short detection time, minimum or no false positives & negatives. These ML techniques are more suitable for classical networks as novel attacks may be detected with greater accuracy, precision, and recall and quickly with low false positives. So, the applicability of these parameters in any solution is a suitable research area.**

  Elaboration: This gap highlights the crucial attributes of security solutions, such as accuracy, quick detection, and minimal false positives and negatives. It suggests that ML techniques, while suitable for classical networks, may require adaptation for VANETs. In our thesis, we investigate how ML-based frameworks can achieve high accuracy and low false positive rates in the context of DDoS attack detection, aligning with your second and third objectives.

- **Existing solutions use various feature sets for attack classification, but no fixed one exists. It is worth finding a broad feature set applicable to different situations.**

  Elaboration: This gap indicates that there is no standardized feature set for attack classification in VANET security solutions. It suggests the need for a comprehensive and adaptable feature set that can be applied across diverse scenarios. In your thesis, you can research and propose a versatile feature set that enhances the robustness and adaptability of your ML-based DDoS detection and prevention framework, aligning with your second and third objectives.

4.3 Objectives

Although the overall objective of this proposed system is to provide a suitable machine learning-based security framework that will be able to detect DoS/DDoS attacks in a vehicular environment, we can achieve this by achieving other supplementary objectives. It requires proper background knowledge and an in-depth analysis of various existing solutions. We should compare the system's efficiency with already available solutions. We use parameters like accuracy, detection time, and false-positive for comparison purposes. Therefore, in the present research proposal, we have opted for the research problem of developing a framework with machine learning techniques for vehicular ad hoc networks. The objectives of this proposal are:

1. To analyze the existing DoS/DDoS security mechanisms and feasibility study in VANETs.

2. To design a Machine learning-based detection framework for DDoS in vehicular infrastructure.

3. To design a Machine learning-based prevention/avoidance framework for DDoS in vehicular infrastructure.
4. To validate and compare the performance of the designed framework with the existing frameworks.

4.4 Motivation

Objective 1: To analyze the existing DoS/DDoS security mechanisms and feasibility study in VANETs.

Motivation: DoS and DDoS attacks are increasing and potentially becoming more harmful than the existing ones. These attacks are analyzed to know about the attack methods used by attackers which helps secure future systems. Existing security mechanisms are studied to find loopholes in them. Attackers use these loopholes or vulnerabilities to launch attacks. We must examine Existing DoS/DDoS in the vehicular environment and their solutions because it will provide a roadmap to future solutions.

Outcome: The primary outcome of the first objective will be a review paper and research gap. This research gap makes a blueprint for future research.

Objective 2: To design a Machine Learning based detection method for DDoS in vehicular infrastructure.

Motivation: Novelty and high performance of machine learning techniques make it a favorite. These AI-based techniques have become more robust with the convergence of deep learning algorithms. We check various parameters like accuracy, precision, recall, and f1 score in ML techniques. Confusion matrix and other tools available in machine learning make them more suitable. There is flexibility in implementing these techniques, like statistical techniques, that we effortlessly combine with ML techniques. All these characteristics make machine learning the first choice of most researchers.

Outcome: The Principal outcome of this objective will be a framework that can detect DoS/DDoS attacks with accuracy and efficiency. This outcome originates from extracted features using a machine learning algorithm.

Objective 3: To design a Machine Learning based prevention/avoidance method for DDoS in vehicular infrastructure.

Motivation: This objective is based on the concept that prevention is better than cure. If we prevent attacks in advance there will be no need to spend time and other resources mitigating those attacks. Prevention also reduces the load of the system by dropping malicious traffic. The Machine learning method has multiple benefits, including those discussed in the motivation section of Objective 2.

Outcome: With the help of this objective, we will generate a DoS/DDoS attack prevention framework that will learn from attack packets. This learning will help in classifying traffic and stopping futuristic attacks.

Objective 4: To validate and compare the performance of the designed framework with the existing frameworks.

Motivation: Any solution is considered acceptable only if outperforms existing solutions. Nobody will use this solution if its performance level is below the existing solutions. Various parameters like accuracy, precision, recall, and f1 score provide the base for the results of solutions. We also use other parameters like throughput, delay, Packet Drop Count (PDC), Packet Transfer Delay (PTD), Packet Transfer Interval (PTI), and waiting time.

Outcome: This objective will provide a comparative statement of results produced by the proposed framework and existing frameworks of the same kind.

4.5 Methodology

In the general research process, we define the problem first and conduct research, followed by other systematic steps. This research implements a security framework using an ML algorithm. We describe a general research methodology in the following steps.

The methodology presented in this study follows a structured sequence of steps, each with a distinct role in the research process. It begins with the crucial task of defining the problem within the relevant domain, ensuring that the research remains focused and purposeful. Following this, a comprehensive review of existing literature is conducted to establish the current state of knowledge and contextualize the research. Identifying research gaps becomes the logical next step, justifying the study by demonstrating its contribution to addressing these gaps. Clear research objectives are then formulated, serving as guiding principles throughout the study. The research design is meticulously prepared to align with these objectives, ensuring that data collection and analysis are systematic and purposeful. Implementation involves executing the research design, collecting data, and subsequently analyzing it to derive meaningful insights. Finally, the interpreted data is compared with existing benchmark results, and the findings are synthesized into a comprehensive report, ensuring that the entire research process is both rigorous and transparent.

1. In the first step, we define the problem related to the concerned domain.
2. After the problem definition, we review the concerned literature.
3. After a review of the literature, we identify the research gap.
4. Based on the research gap, we determine objectives.
5. We also prepare a research design using the objectives.
6. Next, we implement the proposed research design.
7. After implementation, we collect data in terms of its performance.
8. We do the data analysis of this derived data.
9. Finally, we compare the interpreted data with existing benchmark results and write a report.

We described all essential steps in Figure 11, where each step is followed sequentially. After following this process, the four main objectives are formulated. Based on these objectives, the methodology section may be subdivided into different sections.

The methodology begins with problem definition, a crucial step that establishes the research's clear purpose and scope. Following this, a thorough literature review is conducted to gain insight into the existing knowledge landscape, serving as the foundation for subsequent

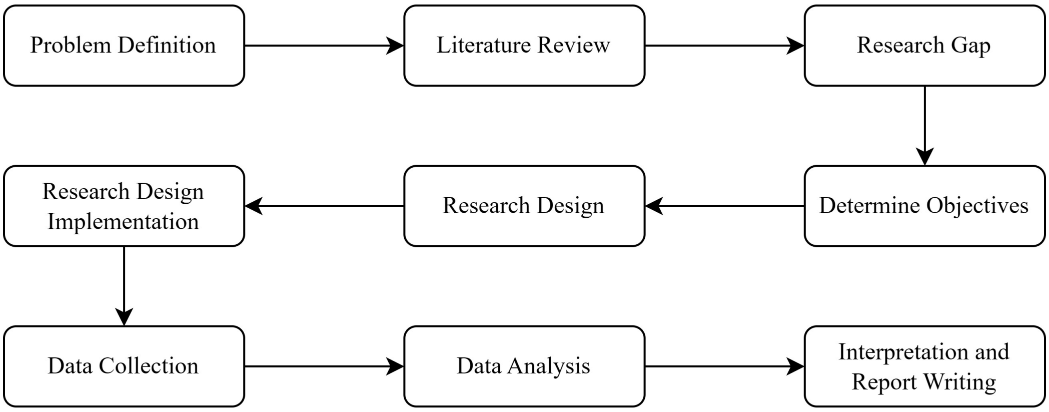

Figure 11: Complete research process

research. The identification of research gaps arises from this review, demonstrating the need for the study. Subsequently, research objectives are formulated to guide the investigation and address the identified gaps. These objectives ensure that the research maintains a clear focus on specific research questions or issues derived from the earlier steps, providing a structured and purpose-driven approach to the study.

4.5.1 Methodology for objective 1: Literature Review

The literature review starts with using books and using search engines. From books, we can find some references. Search engines also provide papers related to that topic. After finding research papers, we may get more links to the concerned subject. This way, related articles, and literature may be retrieved from various sources. Selection of legitimate sources is crucial; otherwise, we may get misleading information.

1. Study about DoS attacks and history, including VANETs.

2. Study about DDoS attacks and history, including VANETs.

3. Study various types of attacks possible in networks, including VANETs.

4. Security solutions, already available to handle DoS/DDoS attacks.

5. Analysis of prevention, detection, response, and tolerance techniques.

6. After analyzing various techniques, a review paper will be completed.

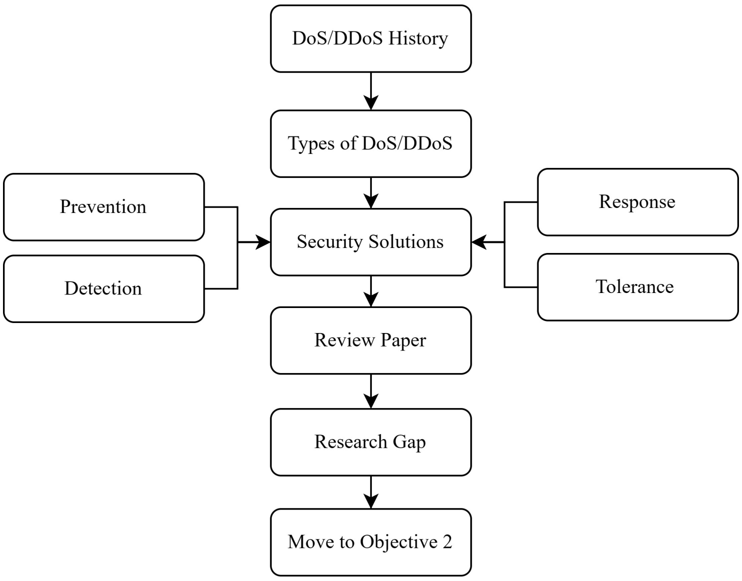

Figure 12: Research methodology for literature review

7. This will lead to research gap identification.

Figure 12 shows the steps involved in the literature review. After finding the research gap, we will lead to objective 2

4.5.2 Methodology for objective 2: Detection framework

In this section of methodology, a novel ML framework will be provided. This framework will be simulated in any simulation environment like MATLAB, SUMO, NS3, or others. Various datasets are available for attack detection, like the NGSIM dataset, KDD CUP 1999, DARPA 1999, DARPA 2000, and CAIDA DDoS 2007, or the learning algorithm may use generated datasets. VANETs will be designed, and traffic will be routed using a suitable protocol. Attacks will be launched by making some vehicles malicious nodes, and data will be captured. From captures of attack traffic, various features will be extracted. Feature selection and extraction are essential steps in selecting the correct feature set. Features like packet drop ratio, packet delivery ratio, packet re-transmission error ratio, packets received, packet capture ratio, packet collision ratio, or packets transmitted may be selected. Features from legitimate traffic will also be extracted. The machine learning algorithm will use both kinds of extracted features for attack detection.

1. In the first step, all the traffic packets will be captured.

2. All the incoming packets will be stored in some database for processing.

3. Stored traffic packets will be used for analysis, and important features will be extracted after analyzing packets.

4. Selected features will be normalized before use by the detection system.

5. After normalization, extracted features will be used by the machine learning algorithm for training and testing. Training and testing models are used to compare the results in the future.

6. After training and testing, the detection system will decide which packets to drop and which to accept for further processing.

7. If packets are malicious, these will be dropped, and we will move further for Objective 3. If packets are legitimate, then packets will be processed.

ML algorithms include support vector machines, random forests, Naive Bayes, IBK, CNN, and ANN. One suitable algorithm will be selected, yielding higher accuracy, precision, f1 score, and recall values. We display the flowchart of the attack detection system with an ML algorithm in Figure 13.

4.5.3 Methodology for objective 3: Prevention framework

In this section, the methodology of a novel attack prevention framework will be provided, which will be based on an ML framework. We display the flowchart of the attack prevention system with an ML algorithm in Figure 14.

1. Input will be taken from objective 2. In case of an attack, malicious packets will be used as input.

2. Probability distribution will also be used as input.

3. In the third step, a binary classifier will be used to classify the malicious and non-malicious packets.

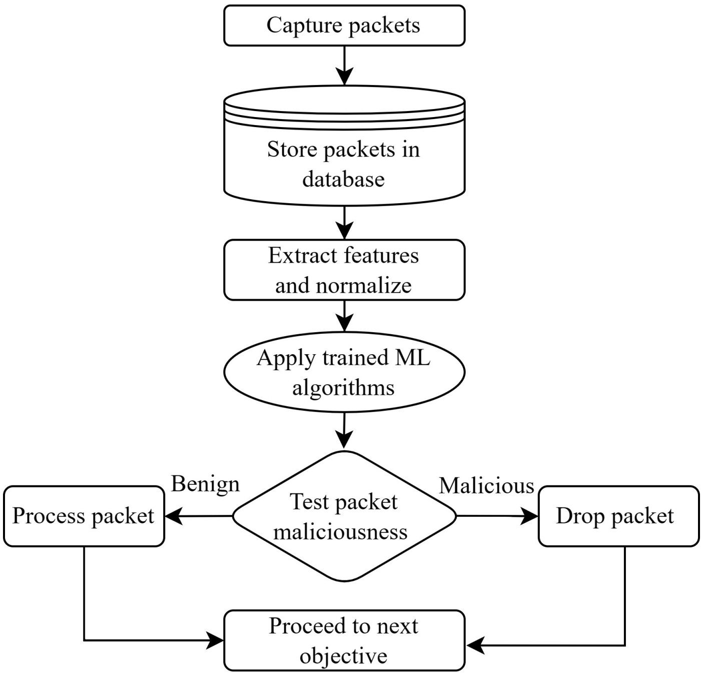
Capture packets
Store packets in database
Extract features and normalize
Apply trained ML algorithms
Benign
Test packet maliciousness
Malicious
Process packet
Drop packet
Proceed to next objective

Figure 13: Research methodology for attack detection

4. In this step Logit model and Probit model will be used to find the probability of suspicious class and non-suspicious class.

5. There may be two cases; in the first case, the probability of a suspicious class may be higher than the non-suspicious one. In this case, packets will be suspicious and will be dropped.

6. In the second case, the probability of no suspicious class may be higher than the suspicious class. In this case, packets will be non-suspicious and processed by the network.

4.5.4 Methodology for objective 4: Validation

This section will evaluate the results obtained by the proposed framework. After the evaluation of the results, these will be analyzed and summarised. After that, these results will be compared with some existing benchmark work with the help of some parameters. Parameters may be accuracy, throughput, delay, or some other parameters. This comparison will provide information about its relative performance. The proposed framework may yield good accuracy, but if it is below the accuracy of existing work, it may not be suitable for

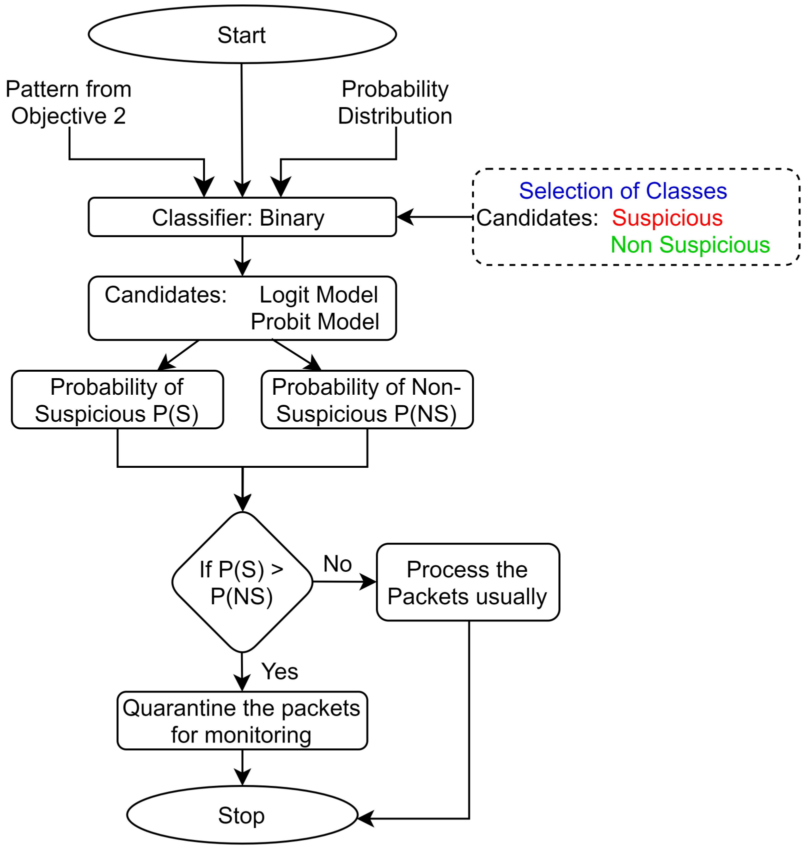

Figure 14: Research methodology for attack prevention

selection. The following Figure 15 shows the steps of objective 4.

The steps involved in this process are:

1. Evaluate the results of the proposed framework.

2. Select some parameters for comparison.

3. Define an efficiency benchmark of some existing work.

4. Compare the results of the proposed framework with the existing one.

5. Make a comparative statement for analysis purposes.

6. Conclude and give remarks about its acceptance or rejection.

4.6 Conclusion

The research methodology chapter explains the framework and strategy to meet the research questions or objectives, an essential part of every research endeavor. This chapter provides

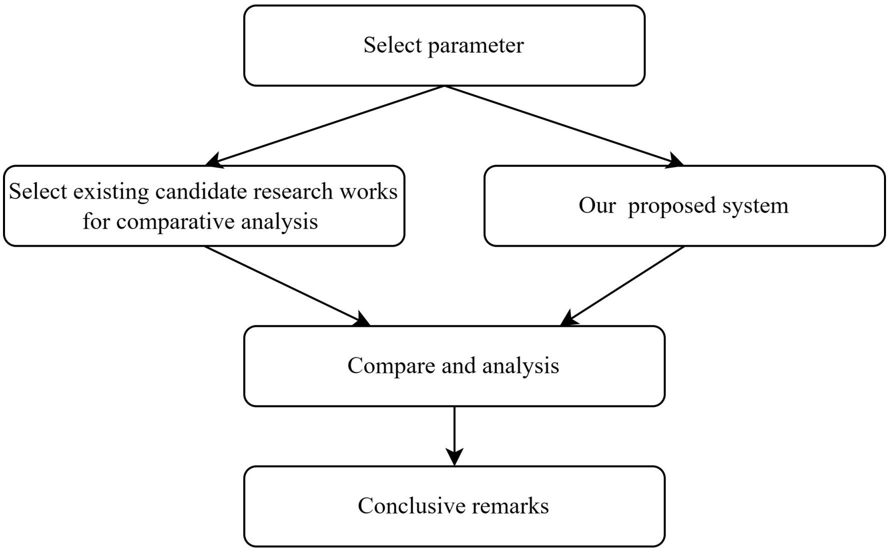

Figure 15: Research methodology for comparative analysis and validation

an overview of the research design, data collection and analysis procedures, and study limitations. The researcher needs to carefully consider the research methodology and select acceptable methods that will adequately address the research questions while preserving the validity and dependability of the findings. The research methods chapter also acts as a manual for other researchers who might want to repeat the study, advancing knowledge in the subject.

Chapter 5

Proposed attack detection framework: VAIDANSHH

This chapter describes the proposed detection framework, VAIDANSHH, for securing VANETs against botnet-based attacks. The chapter starts with an overview of the framework. Next follows an explanation of the purpose and relevance of the proposed models. The experimental setup of the framework is presented, including the system configuration, network topology, and attack methodology. We introduce the Adaptive Alarming Module (AAM) and the Detection Module (DM) and describe them as three tiers. The chapter also includes a comprehensive evaluation of the framework's performance, including the results of tier-1, the hardware level; tier-2, the communication channel level; and tier-3, the application level. Performance metrics for both the AAM and the DM are presented separately. A comparative analysis compares VAIDANSHH with the Naive Bayes family algorithm, other classification algorithms, and existing solutions. Finally, the chapter concludes with a summary of the findings and future work.

5.1 Introduction

This chapter demonstrates a proposed framework for detecting DDoS attacks as VAIDANSHH. Prevention techniques for DDoS attacks are generally proactive and attackers with sophisticated tools can still launch successful attacks. Therefore, it is essential to have robust detection systems that can efficiently detect attacks promptly to minimize losses for the victim and enable early recovery. Intrusion Detection Systems (IDSs) are typically used for this purpose and employ various attack detection strategies. Although IDSs can be implemented in multiple ways, they generally perform the same essential functions of managing, monitoring, detecting, and alarming. The managing component receives and forwards traffic while providing relevant information, which is then analyzed by the monitoring component to identify traffic patterns and other features. The detection component then compares the analyzed results with existing patterns to determine if it is an attack pattern. If so, an alert message is immediately sent to the network administrator. Depending on the organization's requirements and resources, we can categorize DDoS attack detection systems using various approaches, including control mechanisms such as Centralized IDS, Hierarchical IDS, and Distributed IDS, each with advantages and disadvantages [160].

Centralized IDS: In this category of IDSs, alerts are generated by local systems and devices and then forwarded to a central server. This server gathers data from all sources, combines the alerts, and analyzes them to determine the appropriate action. This approach allows for better decision-making due to the collection and analysis of data. However, applying a better security mechanism to the central server is essential as the entire security system could collapse if attacked. A comparison between the working of Centralized and Distributed IDSs was conducted by Feinstein [161].

Hierarchical IDS: In the Hierarchical IDS, we divide the whole system into different levels based on the system's geographical location, operating system, or control authority. Data packets are examined at each level, and alerts are generated and verified by higher-level systems. The higher-level systems act as intrusion detectors and correlation handlers, receiving and generating alerts and analyzing them before forwarding them to higher-level systems. This way, top-level systems take decisions and actions, making the system more scalable. Sharma and Ahmin also utilized Hierarchical IDS in their studies, employing Decision Tree and Rules-Based Models and Machine Learning and Knowledge Models, respectively. (Source: [162] and [30])

Distributed IDS: This defense mechanism distributes overall control among autonomous systems rather than relying on one system. Each component detects and prevents attacks at its level but cannot correlate alerts from other systems, which may result in undetected attacks. Distributed attack detection systems are flexible and easily scalable. One proposed solution for DDoS attack detection is a distributed defense system that employs DIDS with Blockchain and cloud computing infrastructure [31]. We can deploy IDS as host-based or network-based, with hybrid IDS combining the beneficial features [163]. HIDS installed on servers, observe the operating system and other applications working. They investigate data traffic moving to and from the machine and generate alerts for any suspicious activity, such as a process attempting to access the password database. Some HIDS can take on different forms, such as file integrity and login logs [164]. File integrity software is installed when an operating system is installed and maintains a database of binary files. Any changes to these files indicate an attack and alert network administrators. Association rule-based solutions have been proposed for HIDS [165] and logistic regression-based solutions that employ various machine learning techniques [166].

Network IDS: Network-based Intrusion Detection Systems (NIDS) are passive systems or devices installed at selected network points that continuously monitor data traffic. These systems can be implemented using different software or hardware. Generally, NIDS has two interfaces; one for receiving incoming data packets and the other for reporting and controlling. The listening component of NIDS analyzes the patterns and features of incoming data to determine if an attack pattern exists. NIDS are usually deployed near the firewall or edge routers where listening and monitoring network traffic would be more convenient. Various technologies, such as artificial neural networks and machine learning, are employed to inspect traffic patterns. However, large volumes of data cannot be handled by NIDS, and it cannot analyze encrypted packets. Furthermore, Besharati [167] proposed a NIDS-based system that uses logistic regression cost functions with stochastic gradient descent and simulated annealing to fine-tune various hyperparameters of the neural network-based NIDS classifier. Subba [29] also discussed various neural networks and deep learning techniques for NIDS and HIDS. These advancements have enabled fuzzy logic-based systems to predict up to 99.99% of attack patterns in advance and can provide warnings against these attacks.

Hybrid IDS: Hybrid IDS systems utilize Host IDS and Network IDS functionalities to create a more flexible and powerful detection system. By monitoring machine events and network traffic, hybrid IDS systems provide better security and authenticate file system integrity, making them ideal for deployment in highly secure locations like an organization's server. Different deployment options exist when implementing a DDoS defense system, including at the source end, victim end, or intermediate network location. While using a victim-side defense system can provide accurate traffic analysis, it also consumes more resources and is ineffective against high-rate attacks. Employing a hybrid IDS solution using machine learning, such as the one proposed by [168], can help detect DDoS attacks efficiently and accurately.

In this thesis chapter, we propose a machine learning-based solution for detecting DDoS attacks in vehicular networks using tools such as Network Simulator 3, Weka, Python, and Wireshark. The adaptive IDS system proposed in the thesis utilizes sub-components that work together to detect the attack. The information generated by one module is used by the other to improve detection accuracy. The application of the BayesNet algorithm in this process is a novel approach, as it can detect attacks with high accuracy while requiring minimal overhead. Overall, the thesis demonstrates that we can detect DDoS attacks in VANET using a limited number of packet attributes.

- We evaluate the effectiveness of the BayesNet algorithm in a vehicular environment.
- A new dataset is created and utilized in the model, which covers three different scenarios where flooding attacks occur on the unique topology of VANET.
- This model employs a botnet-based attack model that follows a peer-to-peer architecture.
- We implement an Adaptive Alarming Module (AAM) with six distinct features to detect abnormalities in the system, and it also utilizes all or a subset of these features as indicators.
- We also use a unique feature set containing ten features in the detection module. The detection module uses these ten features to identify potential attacks.

5.2 Motivation and contribution

Existing solutions have tackled the problem of DDoS attacks in vehicular environments, but they use only a single-layer architecture for detecting attacks. This architecture relies on parameters from a single tier, resulting in less promising outcomes. To address the limitations of previous studies, we propose VAIDANSHH, an ML-based solution for detecting

DDoS attacks in VANETs. VAIDANSHH is the first model to consider a multi-tier architecture and adaptive traffic thresholds, enabling efficient DDoS attack detection in vehicular environments. VAIDANSHH contributes the following:

- **Three-tier architecture:** VAIDANSHH offers a three-tier security architecture that includes a hardware tier (physical tier), an interface tier (communication channel), and an application tier. This architecture applies rigorous security checks, creating a safe vehicle environment.
- **Adaptive IDS:** Unlike existing static solutions, VAIDANSHH is a dynamic IDS that collects real-time packet information and adapts traffic thresholds according to the traffic load. This adaptation generates more accurate and reliable results.
- **Heterogeneity:** VAIDANSHH is flexible enough to incorporate vehicles of different vendors, standards, protocols, and technologies. As the traffic in VANETs is heterogeneous, vehicles' heterogeneity does not create compatibility or interoperability issues.
- **Dataset:** As DDoS datasets in VANET scenarios are rare, we have generated a new VANET-based DDoS dataset that includes records of benign traffic and DDoS attack traffic of UDP and TCP flood attacks. We can accept this dataset universally as it includes various attack types.
- **Classification:** We pioneer using the BayesNet classification algorithm in the vehicular environment to detect DDoS attacks. BayesNet provides an approach to deal with missing data and enables data to be combined with domain knowledge. The algorithm promotes learning about links between variables, prevents data overfitting, and can forecast accurately even with small sample sizes. Additionally, BayesNet can be easily coupled with decision analysis tools to improve data analysis and management in ML.

5.3 Proposed model: VAIDANSHH

VAIDANSHH is a Network Intrusion Detection System (NIDS) that incorporates three security tiers and uses adaptive traffic thresholds and minimal packet features for attack detection. This innovative layered approach enhances the efficiency and effectiveness of the IDS by reducing false positives, providing multiple layers of protection, and allowing for more advanced incident response. Among the three security tiers, the first tier monitors the hardware resource consumption like CPU and RAM to identify abnormalities. The second tier checks the flow parameters of the communication channel. It identifies attack patterns in traffic, while the third tier utilizes machine learning to detect attacks at the application level. By having security checks at all three tiers, VAIDANSHH is a highly efficient and effective attack detection framework. VAIDANSHH can be deployed at an RSU or TA to monitor VANET traffic because these locations have more computational power and memory to handle large data bursts. Figure 16 shows the overall architecture of VAIDANSHH.

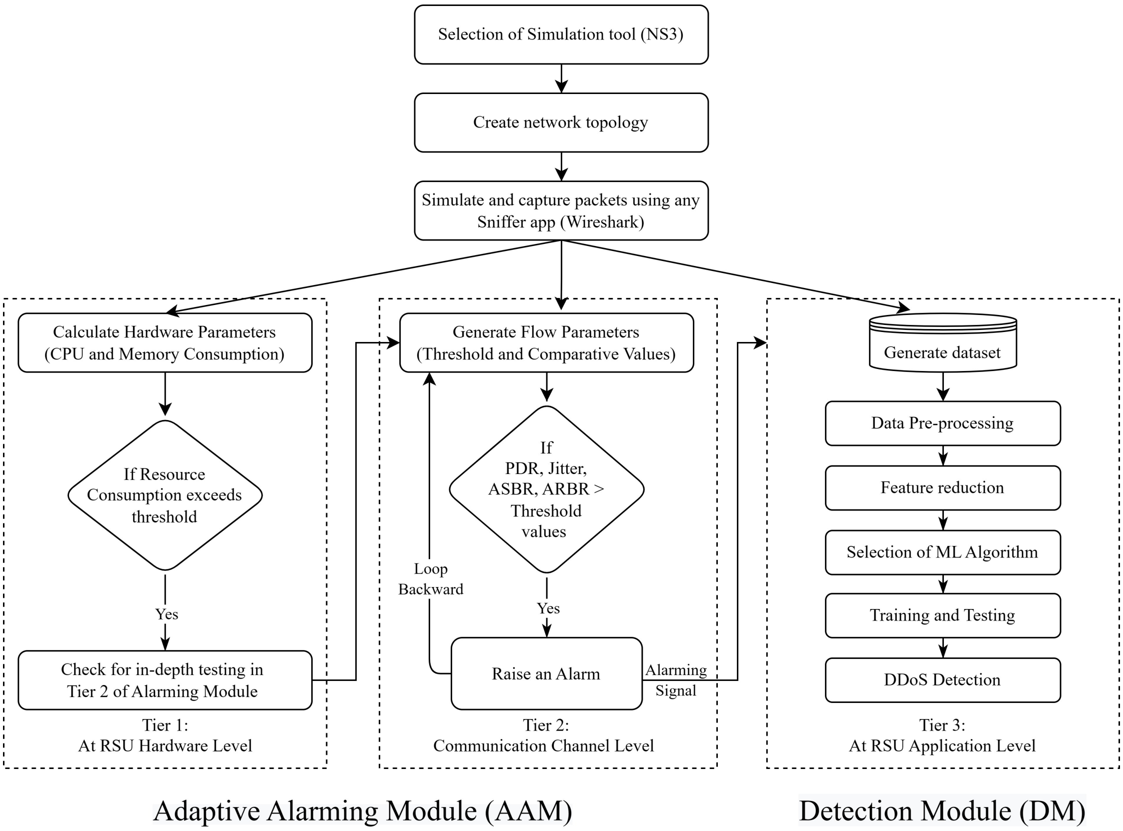
Selection of Simulation tool (NS3)
Create network topology
Simulate and capture packets using any Sniffer app (Wireshark)
Calculate Hardware Parameters (CPU and Memory Consumption)
If Resource Consumption exceeds threshold
Yes
Check for in-depth testing in Tier 2 of Alarming Module
Tier 1: At RSU Hardware Level
Generate Flow Parameters (Threshold and Comparative Values)
If PDR, Jitter, ASBR, ARBR > Threshold values
Loop Backward
Yes
Raise an Alarm
Alarming Signal
Tier 2: Communication Channel Level
Generate dataset
Data Pre-processing
Feature reduction
Selection of ML Algorithm
Training and Testing
DDoS Detection
Tier 3: At RSU Application Level
Adaptive Alarming Module (AAM)
Detection Module (DM)

Figure 16: Overall functionality of proposed system

We divide the complete detection process into two main modules. We describe these 2 modules and the complete functionality of this Figure 16 in the following sections and subsections.

5.4 Experimental setup

The experimental setup for VANET simulation involves creating a virtual environment that emulates real-world scenarios to test the performance of different VANET protocols and applications. We define the vehicles' mobility models, determining how vehicles move within the network. We implement and test the application and then run the simulation to generate results. Different parameters, such as packet delivery ratio, delay, and throughput, are measured and analyzed to evaluate the performance of the VANET protocols and applications. We adjust the experimental setup to test different scenarios. Overall, a well-designed experimental design is essential for accurately evaluating the performance of VANET protocols and applications before deploying them in real-world scenarios.

5.4.1 System configuration

The system configuration used in this study involves several software tools, including Windows 11 as the primary operating system, Oracle Virtual Box Manager, Ubuntu 20.04.2 LTS, Network Simulator 3 (NS3), Weka, Python, and Wireshark. The hardware configuration comprises a specialized 4GB graphics card, an Intel i7 9th generation processor, 8GB of RAM, 256GB SSD, and 1TB HDD. This hardware is suitable for simulating realistic scenarios and experimenting with multiple nodes. The combination of software and hardware tools provides the necessary computational power and storage capacity to conduct simulations with a high degree of realism, making the research results more reliable and accurate.

5.4.2 Network topology

Our simulation uses a specific vehicular topology consisting of 12 nodes, 1 RSU, and 11 legitimate vehicles. We use the terms ”nodes” and ”vehicle” interchangeably. In this topology, one node acts as the attacker, one as the victim, and the other ten as ”bot nodes.”

We illustrate the vehicular topology in Figure 17, where the red car represents the attacker node, the green car represents the victim node, and the orange cars represent the bot nodes. We utilize routing protocols such as UDP, TCP, and ICMP and set the data rate to 512 kbps for benign traffic and 20480 kbps for DDoS traffic, with a maximum bulk data rate of 100000 kbps. The simulation runs for 40 seconds, and the communication area covers 10,000 square meters. To simulate a realistic environment, we apply a random mobility model to all nodes, causing them to move unpredictably within the specified region. We summarize the experiment's environmental parameters in Table 12.

Table 12: Attributes of experimental environment

Parameter	**Value**
Simulation Platform	NS3.2.7
No. of Vehicles	12
Attacker Nodes	1
Bot Nodes	10
Victim	1
No. of RSUs	1
Routing Protocol	UDP, TCP, ICMP
Visualisation Tool	NetAnim
DDoS Rate	20480 kbps
Normal data rate	512 kbps
Maximum Bulk Bytes	100000 kbps
Simulation Time	40 Seconds
Data Transmission Rate	100 Mbps
Communication Range	100 m X 100 m
Mobility Model	Random mobility

5.4.3 Attack methodology

The attack scenario involves two simultaneous attacks. The first is a Vehicle-to-Infrastructure (V2I) attack, which affects the infrastructure because the RSU is kept busy handling malicious messages. The second attack is a Vehicle-to-Vehicle (V2V) attack where one malicious bot node sends a large amount of traffic to the victim vehicle, depleting the victim node's resources. The attacker takes advantage of the topology of small distances to make the attack more lethal. We use peer-to-peer botnet architecture for experiments. We generate a flooding attack (UDP and TCP) using a botnet. The attacker uses indirect communication methods and injects malicious code into vehicles, compromising them. These compromised nodes communicate to form a botnet, which works together without knowledge of each other, controlled by the attacker. The attacker initiates the attack by sending commands to the bot nodes, which send large amounts of data toward the target through the RSU. The attacker considers this architecture more reliable because it is challenging to close an attack from it. Multiple layers of communication easily hide the attacker during the attack. The research aims to detect this type of attack with various layers of security [200]. We simulate three scenarios to incorporate packet features of all types. The first scenario is the Normal Scenario, where vehicles communicate with each other without any attack, and there is only benign traffic. The second scenario is the Attack-only scenario, where only a flooding attack is created without legitimate traffic. This scenario evaluates the system's capacity and performance under a heavy influx of incoming traffic. The third scenario is the Mix Scenario, where a DDoS attack is included with benign traffic. UDP and TCP floods are the two types of flooding attacks in this scenario. Figure 17 depicts the DDoS scenario in the simulation.

As discussed in the previous section 5.2, one unique feature of our model is that it can operate with different types of vehicles that have different communication standards and protocols. In real-world traffic, electric or fuel-powered vehicles may participate in communication, and these vehicles can use unique components. These components may create compatibility issues with elements of other automobiles. Our proposed solution works very well with all these vehicles and accommodates heterogeneous vehicles.

5.5 Adaptive alarming module: preliminary understanding

Some terms need prior explanation before understanding the workings of AAM. These terms are threshold values, comparative values, and adaptivity which are as follows:

Threshold values: Threshold values define the maximum acceptable values for flow parameters. If the parameter values exceed these thresholds, it indicates that there may be some suspicious activity happening in the network. To calculate the threshold values, the AAM considers only legitimate traffic in the network. The AAM calculates the threshold values

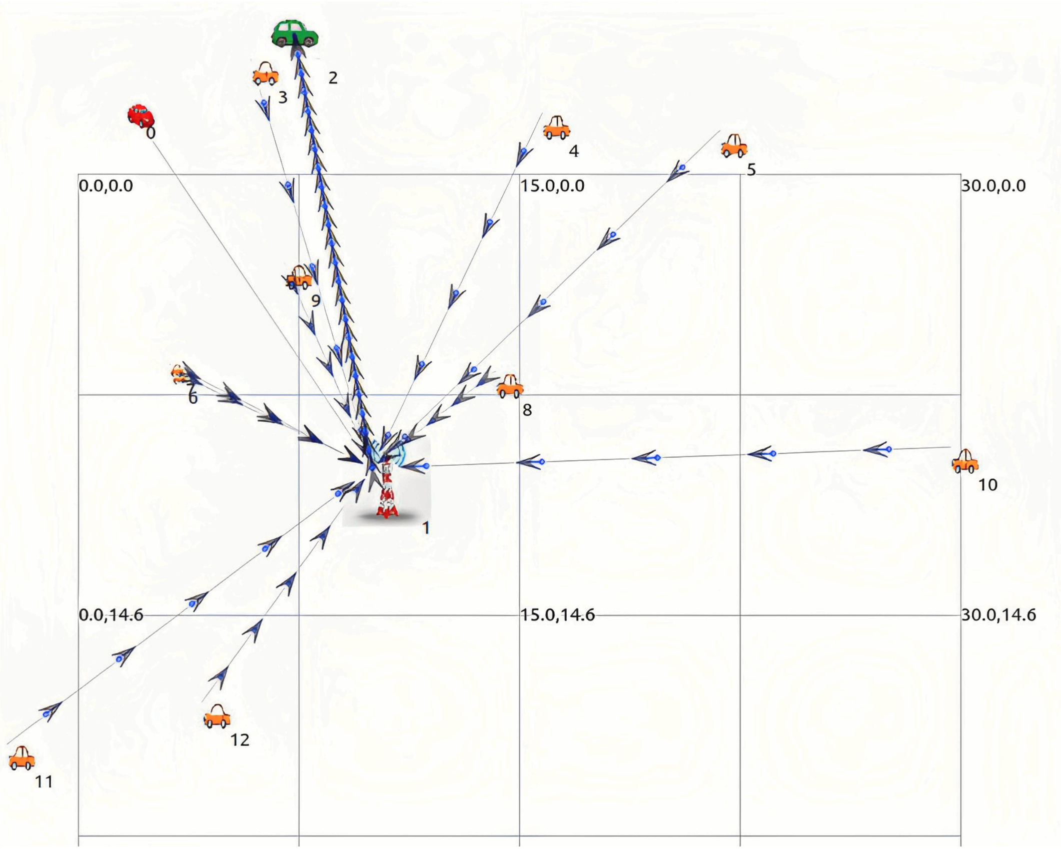

Figure 17: Vehicle set up in the network

initially when we implement the system. We adjust these threshold values dynamically at regular intervals, such as hourly, daily, weekly, or monthly. The AAM model quickly detects malicious traffic that enters the network by regularly recalculating the threshold values. In this experiment, we selected the flow parameter values of normal traffic during peak load. It implies that flow parameter values of normal traffic are treated as base and any value crossing this limit will be suspicious.

Comparative values: The system identifies abnormal flow parameter values of the current traffic, which are then compared against threshold values. The AAM model calculates these comparative values at specific time intervals. These time intervals are shorter than the time intervals of the threshold values. If a comparative value exceeds the corresponding threshold value, the system generates an alert to notify the users of the anomaly.

Adaptivity: A distinctive characteristic of this module is its adaptability in changing threshold values based on traffic patterns. Since the traffic flow may differ at different times, we adjust the threshold traffic parameters based on current traffic conditions. Increased traffic rates usually increase the possibility of DDoS attacks, so the threshold values change dynamically to reflect the current circumstances. In cases such as flash-crowd, where there are sudden surges in legitimate traffic, the module recalculates the threshold values to account for these spikes.

The concept of adaptive thresholds in VANET traffic refers to the process of adjusting threshold values based on current traffic conditions and network characteristics. This process involves monitoring and collecting various traffic parameters like vehicle speed, acceleration, density, and inter-vehicle distance. Statistical analysis is then performed on the collected data to determine the normal range of values for each parameter under typical traffic conditions, using measures like mean, standard deviation, and percentiles. These calculations establish threshold values that define the upper and lower limits of normal parameter values. These thresholds are dynamically adjusted in real time to adapt to changing conditions, taking into account factors like time of day, location, weather conditions, road type, and historical data. The adaptive thresholds account for varying traffic patterns and environmental influences by incorporating these factors. Once the thresholds are established and dynamically adjusted, they serve as reference points to detect abnormal traffic patterns.

Abnormal traffic patterns can include sudden changes in vehicle speed, abrupt acceleration or deceleration, excessively close following distances, increased traffic, or any behavior that significantly deviates from the established normal range. When the current parameter values exceed or fall below the adaptive thresholds, it indicates the presence of suspicious behavior. The system generates warnings or alerts and communicates with nearby vehicles or the centralized traffic management system upon detecting abnormal patterns. These warnings allow for timely responses to potential hazards or abnormal situations on the road.

5.5.1 Functional modules of the AAM

The AAM module is the first component of VAIDANSHH that examines hardware resources and flow parameters to generate alerts. We recommend deploying the AAM in either an RSU or a central TA for optimal efficiency, as in-vehicle components are not equipped to handle heavy traffic or perform complex computations. The AAM has two layers of security, known as Tier-1 and Tier-2. Tier 1 utilizes RAM and CPU utilization as the primary parameters to detect abnormalities in the network. On the other hand, Tier 2 uses flow parameters such as average bit rate sent and received, packet loss ratio, average delay time, jitter, and flow IDs to identify anomalies. During a DDoS attack, there is a significant increase in hardware resource consumption, and flow parameters such as packet loss ratio, delay time, and bit rates also increase. The AAM sets threshold values for these parameters and compares them with the current traffic values to detect any abnormalities in the network. If the hardware resource consumption or the current traffic values exceed the threshold values, it may indicate a DDoS attack, and the AAM raises an alert by sending a message to DM. A more detailed description of the two tiers is provided below:

Tier-1: Hardware level The first layer of our proposed security system focuses on the physical level, specifically the hardware level. It uses CPU and memory utilization as the main parameters to detect abnormal network behavior. Under normal network traffic, hardware resource consumption remains relatively low. However, during an attack, traffic increases, resulting in an abnormal increase in resource consumption. Excessive resource consumption can lead to packet drops and queuing delays due to resource unavailability, ultimately resulting in a Denial of Service (DoS) attack. Our experiments with VAIDANSHH have revealed that the CPU load can increase to its maximum limit of 100%, and memory consumption can exceed 90%.

Tier-2: Channel level In our proposed model, the communication channel serves as the second layer. This layer actively monitors and calculates flow parameters in real-time to identify any abnormalities in the network. As data travels from source to destination, it generates various patterns that help in understanding network traffic behavior. The second layer closely observes these patterns and calculates flow parameters such as jitter, delay, and packet drop ratio. By setting threshold values for these flow parameters, the second layer actively compares them with the current traffic to detect anomalies. If the comparative values exceed the set threshold, it indicates an abnormality and an alert is raised immediately. The AAM generates alarms when it detects any unusual or abnormal activity that may indicate a DDoS attack. These alarms are typically generated in real-time and can be triggered based on various conditions, such as unusual patterns in network traffic or a high number of requests from specific IP addresses or groups of IP addresses. The AAM generates an alarm when the comparative values exceed the threshold values. After generating the alarm, the detection system comes into action.

5.6 Detection module and Tier-3

Our proposed model includes the DM as its second primary module, which uses the BayesNet ML algorithm to detect DDoS attacks. The DM operates at the application level of RSU/TA, which is known as tier-3 in VAIDANSHH. The DM initiates its work once it receives alerts from tier 2 of AAM. The DM uses the BayesNet ML algorithm to analyze network traffic patterns and distinguish between malicious and benign packets. The DM utilizes the Waikato Environment for Knowledge Analysis (WEKA), which aids packet observation, validation, and classification. WEKA examines network traffic patterns and identifies malicious or benign packets.

Additionally, the DM employs a synthesized dataset generated from various simulation scenarios in NS3. We refine this dataset and convert it to a suitable format using data preprocessing and feature reduction. We retain only pertinent features and discard irrelevant ones. After feature reduction, we apply the BayesNet ML algorithm to determine which packets are malicious and which are benign. Following appropriate training, the system creates a model to detect attack packets. The DM tests the dataset to locate DDoS attack packets. We provide a more detailed explanation of the DM's operation in the following sections, which include information on dataset generation, feature extraction and reduction, training, and testing.

5.6.1 Dataset generation:

We have two options for selecting a dataset for machine learning: we can either utilize an existing publicly available dataset or create our dataset by collecting data from various sources. The choice of the dataset will depend on the specific problem we are attempting to solve and the type of model we plan to construct. Some important considerations include the dataset's relevance, size, quality, and whether it is labeled or not. For our experiment, we decided to use a self-created dataset based on possible attack scenarios discussed in subsection 5.4.3, which was designed specifically for attack detection and contains an adequate number of features. Three scenarios are simulated in NS3 i.e., a Normal traffic scenario, an attack traffic scenario, and a mixed traffic scenario. Using NS3, we monitored packet movement using a flow monitor among 12 nodes and saved relevant packet attributes in a .csv file by opening the "pcap" file in Wireshark. Packet data from all three scenarios is captured in a single file making our synthetic dataset. Our synthetic dataset contains 28 attributes and 6,97,792 records.

5.6.2 Feature extraction and dimension reduction

The DM uses a supervised learning technique called CFS Subset Evaluator and the best first search method in WEKA to pick features. CFS evaluates the feature subset by calculating the correlation coefficient between each attribute and the class attribute, as described in [201]. The evaluator identifies the most important attributes for classification and may choose fewer attributes than those specified in the evaluator's settings. It then combines the selected features in a subset and calculates the correlation coefficient between the subset and the class attribute. The feature subset with the highest correlation coefficient is considered the best subset. We use the CFS Subset Evaluator in machine learning to identify and select the most relevant attributes from a given dataset. It operates by first computing the individual correlation of each feature with the target variable, typically a class label. This step is crucial in assessing how well each feature independently predicts the target. Features that exhibit higher correlations are considered more informative for the task at hand, as they carry valuable information. In our case, the class attribute classifies the packet into normal, attack, or mixed class. However, CFS doesn't stop at individual feature correlations; it also takes redundancy into account. Redundancy refers to the extent to which two or more features convey similar information. We choose the following ten attributes as shown in Table 13 from a total of 28 attributes.

Table 13: Selected attributes of synthetic dataset

Attribute	**Description**
Time	Indicates the time of packet sent
Source	IP address of the sender.
Destination	IP address of the receiver.
Protocol	Protocol used for communication.
Stream Index	It maps the source and destination port of two IP addresses.
Time to Live	Time for which a packet should remain in the network.
Time Since Previous Frame	Time since previous frame was transmitted in the network.
Arrival Time	Time when the packet arrived at its destination
Epoch Time	Time in seconds used by Wireshark for pcap files.
Info	Information about packets sent.
Class	Attribute for labeling the packet type.

Table 13 exhibits the chosen attributes of the generated dataset of the proposed system. In addition, we perform feature reduction using nine distinct algorithms: five ranker methods, three greedy step-wise search methods, and one best first search method [202] to check which attributes are most suitable and relevant. Ten attributes with one additional attribute i.e., the class attribute are used in classification.

5.6.3 Training and testing

We use BayesNet, for training and testing for DDoS attack detection, which is a powerful machine learning algorithm utilized to model and analyze the probabilistic relationships between network traffic features. In this application, we train the algorithm on historical network data that includes both legitimate traffic patterns and instances of DDoS attacks. During the training phase, BayesNet learns the conditional probabilities and dependencies between features, building a Bayesian network that represents the network's behavior under normal and attack conditions. Subsequently, we use the trained BayesNet model to assess incoming network traffic in real time. By comparing the observed behavior of the network to the learned probabilistic model, BayesNet can detect deviations that suggest the presence of a DDoS attack. This framework offers a probabilistic and interpretable approach to DDoS detection, enhancing network security by identifying and mitigating malicious traffic effectively.

We use various data splits to train and test our model and choose the split that produces the highest classification accuracy. Our synthetic dataset contains 6,97,792 instances, which we split into proportions to determine the optimal split. We assess the model's performance using several metrics: time taken, accuracy, true positive rate (TPR), false positive rate (FPR), precision, recall, and f-measure. After comparing the results, we find that the second case, where 40% of the data is used for training and 60% for testing, is the most suitable split. This split yields a high accuracy rate of 99.9785% and fast testing speed. The total time to train the model is 2.84 seconds, and testing takes 1 seconds. The results from the training and testing are presented in Table 14.

Table 14: Training and testing results with different data splits

BayesNet	**30% Training 70% Testing**	**40% Training 60% Testing**	**50% Training 50% Testing**	**60% Training 40% Testing**	**70% Training 30% Testing**
Time to Build Model in Seconds	3.44	2.84	2.83	2.66	2.94
Time to Test Model in Seconds	1.53	1.00	0.87	0.77	0.55
Instances Tested	488454	418675	348896	279117	209338
Correctly Classified	488355	418585	348828	279065	209298
Correctly Classified %age	99.9797	99.9785	99.9805	99.9814	99.9809
In Correctly Classified	99	90	68	52	40
In Correctly Classified %age	0.02	0.02	0.01	0.01	0.01
Mean Absolute Error	0.00	0.00	0.00	0.00	0.00
Root Mean Squared Error	0.01	0.01	0.01	0.01	0.01
Relative Absolute Error	0.04	0.04	0.04	0.04	0.42
TP Rate	1	1	1	1	1
FP Rate	0	0	0	0	0
Precision	1	1	1	1	1
Recall	1	1	1	1	1
F-Measure	1	1	1	1	1
MCC	1	1	1	1	1
ROC Area	1	1	1	1	1
PRC Area	1	1	1	1	1

5.6.4 DDoS detection

The BayesNet algorithm has two stages: structure learning and parameter learning. In the structure learning stage, the algorithm identifies the variables' interdependence. It starts with an empty network and gradually adds edges, evaluating the score of each new structure. The algorithm chooses the structure with the highest score and repeats the process until we can not improve further. The resulting network structure is the one with the highest score. In the parameter learning stage, the algorithm determines the probability distributions of each variable given its parent. Methods such as maximum likelihood estimation or Bayesian estimation can be employed for this purpose. Once the structure and parameters of the network are learned, the network can make predictions regarding the probability of each variable based on the values of other variables.

ML detection also utilizes the K2 search method. This method assigns scores to various network structures based on the probability of the data given the structure. It also assigns scores to a set of input variables in random order. The order of variables plays a critical role in the algorithm's performance, and an incorrect order can lead to incorrect learning of network topology. Once the network structure is determined, the DM chooses a class in the estimate package to determine how to learn the probability tables. In this model, we use the Simple Estimator Class (SEC), which directly estimates conditional probabilities using Equation 1.

$$P(x_i = k | pa(x_i) = j) = \frac{N_{ijk} + N'_{ijk}}{N_{ij} + N'_{ij}}. \tag{1}$$

Equation 1 uses a default value of 0.5 for the alpha parameter, denoted as N'_{ijk}. This default value balances the trade-off between over-fitting and under-fitting and is often a good starting point for most applications, with fine-tuning possible later. The detection model incorporates filters such as replace-missing-values, string-to-nominal, and remove-percentage-split. The model uses ten characteristics and a dataset of 418,675 instances. The training set is built in 2.84 seconds, and the detection functionality is tested in 1 second. When the DM checks all instances and identifies malicious packets, it discards them, and only benign packets are processed. The VAIDANSHH model efficiently detects DDoS attacks, and its complete working is defined in Algorithm 1.

Algorithm 1 Algorithm of VAIDANSHH

```
procedure ALARM()
    $interval1 \leftarrow time1$;
    $interval2 \leftarrow time2$;
    $stTimeThr \leftarrow time3$;
    $stTimeComp \leftarrow time4$;
    procedure TIMER(interval2, stTimeThr)
        $thr = calculateThreshold()$;
        $return(thr)$;
    endprocedure
    procedure CALCULATETHRESHOLD()
        PythonScript();
    endprocedure
    procedure TIMER(interval, stTimeComp)
        $comp = calculateParameter()$;
        $return(comp)$;
    endprocedure
    procedure CALCULATEPARAMETER()
        $PythonScript()$;
    endprocedure
    if $comp \geq thr$ then
        raise alarm;
        DETECT();
    else
        process packets;
        continue();
    end if
endprocedure
procedure DETECT()
    pre-process data;
    select features;
    select ML algorithm;
    train the ML model;
    use dataset;
    if $Packet = Malicious$ then
        sinkhole malicious packets;
        update log;
    else
        process packet;
    end if
endprocedure
```

Our model has two procedures, namely $alarm()$ and $detect()$, which are relevant to our work. The operation of the AAM is defined in Section 5.5.1 and Section 5.5.1. The $alarm()$ procedure is responsible for the AAM's functioning, which uses four variables to define the start and interval time of the procedures and the time interval when the system calculates the threshold and comparative values. A timer procedure is initiated at the specified start time and repeats at the predetermined interval. The timer procedure calls the $CalculateThreshold()$ and $CalculateParameter()$ procedures, which utilize a Python script to compute and return threshold and parameter values. The system then compares these values, and if the comparative value exceeds the threshold value, the $alarm()$ procedure sends an alert message to the RSU controllers or network security administrators. The administrators can take immediate action or conduct a further diagnosis.

5.7 Results and discussion

In this section, we discuss the performance of our proposed VAIDANSHH model in detecting DDoS attacks. We evaluated our solution on a synthetic dataset and achieved the following outcomes. The result section covers two subsections. The first subsection, labeled as 5.8, outlines the evaluation metrics used to measure the model's effectiveness. The second subsection, labeled as 5.9, presents the results of all three tiers and illustrates them through tables and figures.

5.8 Performance metrics

Various performance metrics are employed to assess the efficiency of DDoS attack detection systems. These metrics include detection rate, false positive rate, false negative rate, throughput, latency, accuracy, and false alarm rate. Different metrics are utilized in the AAM and the DM to evaluate their performances.

5.8.1 AAM Metrics

We assess the effectiveness of the AAM using six different metrics: the average ratio of packet loss, the average delay in mean, the number of flows, the average bitrate received and sent, and the amount of jitter.

Average packet loss ratio: Regarding network communication, some packets may not reach their intended destination due to network congestion or errors. It is important to minimize packet loss to enhance the network's performance. Packet Loss Ratio (PLR) is a metric used to calculate the ratio of lost packets to the total number of packets sent, represented as a percentage. The equation below, labeled Equation 2, can be used to calculate the PLR for each flow.

$$PLR = (\frac{n^r - n^s}{n^s}) * 100. \tag{2}$$

where PLR stands for packet loss ratio, n^r for the total number of packets received, n^s for the total number of packets sent. The system calculates APLR, i.e., the average of all flows from the Equation 3.

$$APLR = \frac{(PLR_1 + PLR_2 + PLR_3 + ... + PLR_n)}{n}. \tag{3}$$

Equation 3 calculates the Packet Loss Ratio (PLR) of each flow, where PLR1 represents the PLR of the first flow, PLR2 represents the PLR of the second flow, PLRn represents the PLR of the last flow, n represents the total number of flows. The PLR, also known as the Packet Drop Ratio (PDR), is the ratio of lost packets to the total number of packets sent, represented as a percentage. Networks with higher average PLR (APLR) or average PDR (APDR) indicate poor performance, while those with lower APLR or APDR indicate better performance.

Average mean delay: Delay is a crucial network performance parameter that measures the time taken by a bit to travel from its source to the destination or from one endpoint to another. Longer delay times indicate lower network performance and vice versa. This parameter is also known as transmission delay. To calculate delay, we subtract the time a packet is received from when we sent it. We then sum all the delays of each packet to get the total delay, dividing by the total number of packets to get the average delay. Similarly, we can calculate the Average Mean Delay (AMD) using Equation 4.

$$AMD = \frac{(MD_1 + MD_2 + ... + MD_n)}{n}. \tag{4}$$

Equation 4 provides a way to calculate the Average Mean Delay (AMD), a performance parameter in network communication. We calculate AMD by taking the mean delay of each flow and then averaging them. To calculate the mean delay, we first find the delay for each packet by subtracting the receiving time from the sending time. We then add all the delays for all the packets and divide them by the total number of packets for each flow. The result for all flows is then averaged to obtain the AMD. In the equation, MD1 represents the mean delay for flow 1, MD2 represents the mean delay for flow 2, MDn represents the mean delay for the last flow, and n represents the number of the last flow.

Number of flows: The term ”flow” can have various meanings in different computing contexts, but in this particular case, it refers to a set of data streams or bits transmitted between two nodes during a session. During a DDoS attack, the number of flows of data streams usually increases significantly. However, in computer networks, there may be situations where the number of flows increases without any attack, such as during a temporary increase in traffic due to a flash crowd event. In vehicular networks, however, increased suspicious activity flows require further investigation.

Average received and sent bitrate: The average received and sent bitrate is used to evaluate the impact of a DDoS attack. Bitrate is the amount of data transmitted in a particular time unit, also known as the transfer rate. It is generally measured in bits per second but can also be measured in bytes or kilobytes per second. When a node is under a DDoS attack, it receives a large volume of data and sends limited or no data because it is busy processing incoming packets. This model calculates the average received bitrate using Equation 5.

$$ARB = \frac{(RBR_1 + RBR_2 + ... + RBR_n)}{n}. \tag{5}$$

The equation shown in Equation 5 calculates the model's average received bitrate (ARB). The received bitrate of each flow is denoted by RBR1 for flow 1, RBR2 for flow 2, and RBRn for the last flow, where n is the total number of flows.

$$RBR = \frac{(RBytes * 8)}{RDur}. \tag{6}$$

and

$$RDur = timeLastRxPkt - timeFirstRxPkt. \tag{7}$$

In a similar way average sent bitrate is calculated using the following Equation 8

$$ASBR = \frac{(SBR_1 + SBR_2 + ... + SBRn)}{n}. \tag{8}$$

The preceding equation represents the calculation of the average sent bitrate. Here, ASBR refers to the average sent bitrate, SBR1 denotes the sent bitrate of the first flow, SBR2 represents the sent bitrate of the second flow, and SBRn denotes the sent bitrate of the last flow, where n is the number of the last flow.

$$SBR = \frac{(SBytes * 8)}{SDur}. \tag{9}$$

and

$$SDur = timeLastTxPkt - timeFirstTxPkt. \tag{10}$$

The formula shown in Equation 10 utilizes the variable $SDur$, which represents the time duration taken for sending packets, calculated by subtracting the time of the first packet from the time of the last packet.

Jitter: The variation in the delay of data packets is called packet jitter or delay variation. If packets take the same time to reach their destination, there is no jitter. However, if packets experience different delays, the delay variance is referred to as jitter. From a performance perspective, the jitter should be minimal. To calculate the jitter, we use the average time difference between two successive packets, which can be obtained using Equation 11.

$$TDif_1 = TTFP - RTFP. \tag{11}$$

Where TTFP denotes the Transmit time of the first packet of a particular flow and RTFP Receiving time of the first packet of a particular flow.

$$Jitter = \frac{(TDif_1 + TDif_2 + ... + TDif_n)}{n}. \tag{12}$$

In our developed system, we utilize the average jitter produced by various flows. To obtain this value, we add all the jitter values and divide the sum by the total number of flows.

5.8.2 DM Metrics

The Detection Module (DM) is an essential part of our system, which detects malicious activities within a network. We utilize different metrics to determine the efficiency of an attack detection module. Accuracy, precision, recall, and F1 score are some of the common metrics that are employed to evaluate the performance of an attack detection module.

Accuracy: The accuracy refers to the degree of correctness of our model in predicting the correct packet labels or classes. It is influenced by both the datasets used and the algorithms employed. Normalized datasets tend to yield higher accuracy than non-normalized ones. Additionally, different machine learning algorithms have distinct training, learning, and testing procedures, which lead to different outcomes. To calculate accuracy in machine learning, we use the following Equation 13:

$$Accuracy = \frac{TP + TN}{N}. \tag{13}$$

where the TP is True Positive, TN is True Negative and N is Total number of samples.

Precision: Precision is a metric used to measure the correctness of a model's positive predictions. It is calculated by dividing the number of true positive predictions by the total number of predicted positives. During testing, a machine learning algorithm can predict multiple classes or categories, and each class produces its precision value. The precision metric is particularly useful when accuracy in positive predictions is important, ensuring the model performs well in predicting any given label.

$$Precision = \frac{TP}{TP + FP}. \tag{14}$$

Recall: The recall is a metric representing the proportion of true positives predicted by the model among the total actual positives in the dataset. In other words, recall tells us how well the model can identify all the relevant instances in the dataset. The formula for recall can be expressed as follows:

$$Recall = \frac{TP}{TP + FN}. \tag{15}$$

In Equation 14 and Equation 15, TP, FP, and FN stand for true positive, false positive, and false negative respectively.

F1 Score: When we want our model to have a balanced precision and recall score, we combine them to obtain a single metric. The F1 score is useful in this scenario, as it is the harmonic mean of precision and recall.

5.9 Evaluations

The results obtained from the simulation and proposed model can be divided into three subsections. In the first subsection 5.9.1, we calculate intermediate results for tier-1, which refers to the hardware level. In this part, we consider CPU utilization and memory consumption as parameters. The second subsection 5.9.2 displays the results of tier-2, which relates to the communication channel. It includes parameters such as the average packet loss ratio, mean delay, sent bitrate, received bitrate, number of flows, and jitter. Lastly, the third subsection 5.9.3 presents the results in the context of attack detection. These results are from tier 3, representing the application level, using accuracy, precision, recall, and F1-score as the main parameters.

5.9.1 Results of tier-1

During the analysis of the results, we initially focused on the utilization of hardware resources. Tier 1 handles memory consumption and CPU load. When an attack occurs, the CPU load can reach a maximum value of 100%. The system crashes, and communication ceases in real-world scenarios, such as the simulated attack. We also notice a rise in memory consumption, which reaches a maximum of 1.77 GB during the attack. It is the maximum memory available that is exhausted during the attack.

5.9.2 Results of tier-2

As mentioned previously, VAIDANSHH employs channel parameters to detect attacks in addition to packet information. These parameters are extracted using Python and include the average packet loss ratio, average sent and received bitrate, mean delay, and flow IDs. The results for these parameters are presented in Table 15, with the first column displaying the benign traffic results and indicating threshold values. Table 15 describes the results.

We can draw the following conclusions based on the AAM results. Firstly, the number of flow IDs increases in attack scenarios, indicating abnormal behavior in traffic. Secondly, the packet loss ratio (PLR) increases significantly from 0% to 46%, another indicator of abnormal behavior. Thirdly, the average means delay increases from $1ms$ to $35ms$,

Table 15: Results of alarming module from flow parameters

Flow parameter	**Normal Traffic (Thresholds)**	**Normal Traffic with Attack**	**UDP Flood Only**
Average Packet Loss Ratio (APLR)	0.00	30.87	46.09
Average Sent bitrate (ASB) in kbit/s	0.31	12001.65	18006.16
Average Received bitrate (ARB) in kbit/s	0.31	12001.65	18006.16
Average Mean delay (AMD) in milli seconds	1.02	27.04	35.58
Jitter in milli seconds	0.00	541.26	415.62
Number of Flow IDs generated	5.00	12.00	18.00

suggesting abnormal behavior in the traffic. Fourthly, the average sent bitrate increases significantly from almost $0.5 kbit/s$ to $18007 kbits/s$. The jitter also increases from 0 to 541.26 in the mixed scenario and 415.62 in the UDP flood attack. The detection accuracy is very high at 99.9785%, and the detection process takes only 1 second. The model has a true positive rate of 1 and a false positive rate of 0, which means that the model can accurately identify and drop malicious packets. Additionally, the precision, recall, and F-measure values of 1 indicate that the VAIDANSHH model is efficient regarding these parameters.

5.9.3 Results of tier-3

The module employs several parameters such as accuracy, precision, recall, and f1-score to evaluate the effectiveness of attack detection. However, detecting attacks with high accuracy is the primary objective; hence, accuracy is considered the topmost parameter.

Table 16: Detection results

Parameter	**Value**	**Percentage**
Correctly Classified Instances	418585	99.9785
Incorrectly Classified Instances	90	0.0215
Kappa statistic	0.9995	-
Mean absolute error	0.0001	-
Root mean squared error	0.0119	-
Relative absolute error	-	0.0474

Root relative squared error	-	3.0688
Total Number of Instances	418675	-

Table 17: Confusion Matrix

A	B	C	Classified as
276894	48	0	**A = Mix Flood**
0	2854	0	**B = Normal**
0	42	138837	**C = UDP Flood**

Table 16 presents the accuracy results, while Table 17 shows the confusion matrix of the model. The confusion matrix reveals that among 276942 instances of the Mix Flood class, the model accurately classifies 276894 instances but incorrectly labels 48 instances. On the other hand, all 2854 instances of the legitimate category are correctly classified with no misclassification. For the UDP Flood class, out of 138879 instances, 138837 are correctly classified, while 60 instances are wrongly classified as legitimate class instances.

Table 18: Detailed accuracy by class

TPR	FPR	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
1	0	1	1	1	1	1	1	Mixed
1	0	0.969	1	0.984	0.984	1	1	Normal
1	0	1	1	1	1	1	1	Flood
Average results								
1	0	1	1	1	1	1	1	-

Table 19: Results obtained from techniques of Naive Bayes family

N B Family at 40%-60%	Bayes Net	Naive Bayes	Multinomial	Simple	Updatable
Time to Build Model (Secs)	2.84	0.97	0.08	0.16	0.68
Time to Test Model (Secs)	1.00	2.29	0.36	0.76	1.86
Instances Tested	418675	418675	418675	418675	418675
Correctly Classified	418585	414927	276942	405851	414927
Correctly Classified %age	99.97	99.10	66.14	96.93	99.10

In Correctly Classified	90	3748	141733	12824	3748
In Correctly Classified %age	0.02	0.89	33.85	3.06	0.89
Mean Absolute Error	0.00	0.01	0.30	0.02	0.01
Root Mean Squared Error	0.01	0.06	0.38	0.11	0.06
Relative Absolute Error	0.04	4.11	100	7.72	4.11
MCC	1	0.98	0.99	0.93	0.98
ROC Area	1	1	0.5	1	1
PRC Area	1	1	0.548	1	1

Table 20: Results obtained from other algorithms

BayesNet and other algorithms	**BayesNet**	**J48**	**AdaBoost**	**Ensemble**	**OneR**	**Grading**
Time to Build Model (Secs)	2.84	6.23	16.30	294.20	1.14	5.66
Time to Test Model (Secs)	1.00	0.49	0.50	206.52	0.47	0.77
Instances Tested	418675	418675	418675	418675	418675	418675
Correctly Classified	418585	485112	415817	276807	277740	276942
Correctly Classified %age	99.97	99.31	99.31	66.11	66.33	66.14
In Correctly Classified	90	3342	2858	141868	140935	141733
In Correctly Classified %age	0.02	0.68	0.68	33.88	33.66	33.85
Mean Absolute Error	0.00	0.00	0.00	0.30	0.22	0.22
Root Mean Squared Error	0.01	0.03	0.03	0.38	0.47	0.47
Realative Absolute Error	0.04	0.94	0.94	100.13	74.58	74.00
MCC	1	1	0.99	0.99	0.04	0.99
ROC Area	1	1	1	0.50	0.50	0.50
PRC Area	1	1	1	0.54	0.55	0.54

The outcomes from the additional sections are consolidated in Table 18 and classified into three important parameters. The first row represents the results for the mixed class, where the True Positive Rate is 1 or 100%, False Positive Rate is 0 or 0%, and recall is 1 or

100%. Precision and F-Measure values are also 1. In the same way, the parameters for the normal and flooding classes are displayed in the second and third rows. The final row of the table presents the average results, which are outstanding in all three scenarios, and the variance in the results is not significant, indicating that the VAIDANSHH model is an effective framework for identifying attacks.

5.10 Comparative analysis

This section aims to compare the performance of VAIDANSHH with other ML algorithms applied to both the synthesized and publicly available datasets. The final results are compared based on the parameters discussed in the previous section.

5.10.1 Naive Bayes family

We utilized various algorithmic variations of the naive Bayes technique on the same dataset with different partitions. Our proposed model used a split of 40% training and 60% testing data with the BayesNet algorithm, while the other algorithms used the same partition. BayesNet outperformed other algorithms in terms of correctly classified instances, percentage of correctly classified cases, and incorrectly classified instances, as well as in the percentage of incorrectly classified samples, mean absolute error, root mean squared error, relative absolute error, true positive rate, false-positive rate, precision, recall, f-measure, and MCC. The results are presented in Table 19 and Figure 18.

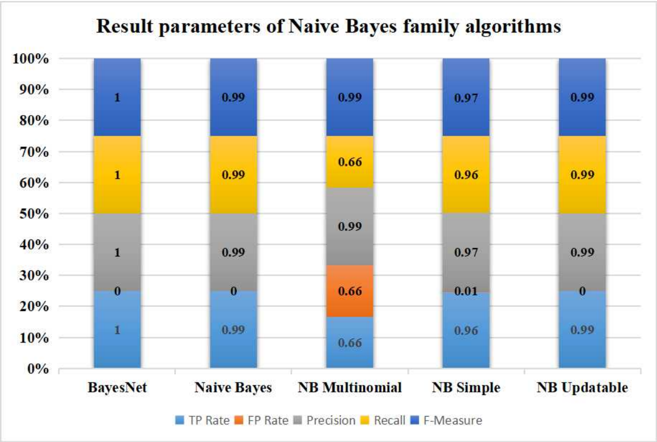

Figure 18: Results obtained from Naive Bayes algorithms

From Figure 18 it is clear that the BayesNet algorithm is providing the best results when compared to other algorithms in the NaiveBayes family.

5.10.2 Other classification algorithms

This section of the results presents the performance of several non-Bayes algorithms. Following evaluating the Bayes family algorithms, we tested other ML classifiers, including J48, Ada-BoostM1, Ensemble Selection, OneR, and Grading algorithms. The results of these algorithms varied, with some performing well and others not. For example, J48 and Ada-BoostM1 achieved high accuracy rates, but their training times were long, and all result values needed to be corrected. Ensemble Selection took a very long time for training and testing, yet its results needed more accurate. OneR and Grading algorithms were fast in training and testing, but their performance could have been more satisfactory. The results of these algorithms are summarized in Table 20 and Figure 19.

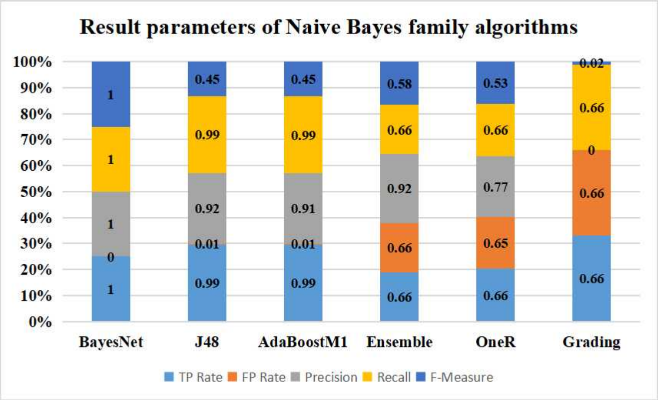

Figure 19: Results obtained from other algorithms

Again from Figure 19 it is clear that BayesNet is superior among other algorithms in terms of TP Rate, FP Rate, Precision, Recall, and F1-Measure.

5.10.3 Existing solutions

In this section of the results, we have compared our model VAIDANSHH with other novel solutions in a similar domain. We have evaluated these solutions based on various parameters, focusing on accuracy. Zang et al. (2021) propose a machine learning-based IDS that uses streaming engines to analyze, handle, and visualize massive data in VANET, with an accuracy of 95% [203]. Another approach for VANET security is a hybrid data-driven model that identifies known network breaches, with an accuracy of 96% [204]. Goncalves et al. (2021) proposed an ML solution to provide an intelligent hierarchical security framework for VANET, with an accuracy of 96% [205]. Soni et al. (2022) introduced a system to secure V-RSU communication from blackhole and wormhole attacks in VANETs, with an accuracy of 89% [207]. Malik et al. (2022) suggest another method for detecting and preventing blackhole attacks in VANET, with an accuracy of 95% [208]. Finally, we report the results of VAIDANSHH, which achieves an accuracy of 99.9%.

Table 21: Comparison of VAIDANSHH with existing models

Reference	**Methodology**	**Detection type**	**Dataset**	**Accuracy claims**
Zang et al. (2021) [203]	Random Forest	Anomaly-based	Mininet-Wifi and CIC-IDS 2017	95%
Bangui et al. (2021) [204]	Random Forest and K-means	Hybrid (Anomaly and Signature-based)	CIC-IDS 2017	96%
Goncalves et al. (2021) [205]	Random Forrest, J48, and Multi-layer Preceptron	Anomaly-based	Public Vehicular dataset available at [206]	96%
Soni et al. (2022) [207]	Particle Swarm Optimization Approach	Behavior-based	Not required, as attacker behavior is observed	89%
Malik et al. (2022) [208]	Detection & Prevention of Black Hole Attacks	Anomaly-based	Not used, as other parameters are used	95%
Proposed VAIDANSHH	BayesNet	Anomaly-based	Synthesised	99.9%

5.11 Conclusion and Future Work

The proposed model for detecting DDoS attacks is based on parameter selection, where the AAM can identify suspicious network activity. Various parameters, such as jitter, delay, and packet loss ratio, can indicate suspicious activities. The DM can detect attacks effectively with an average accuracy of 99.9785, precision, recall, and F1-Score of 1 each. The BayesNet algorithm performs very well in this model, with a weighted TPR of 1 and a weighted false-positive rate of 0, indicating that all packets are accurately classified and legitimate users receive all the services. The proposed VAIDANSHH model is suitable for detecting attacks with high accuracy, almost 100%. With TP rate, FP rate, precision, and recall values of 1 and 0, the proposed model is also appropriate for other parameters. The synthesized dataset used in the model contains the appropriate number of parameters and records, presenting a realistic representation of vehicular networks. Other researchers working on intrusion detection in VANETs can also use this solution to understand the nature of DDoS attacks. Concentration is required for attack prevention to avoid additional processing overheads, and flow parameters can identify suspicious activities and produce good results.

Our future research will be directed toward designing a system that can detect and respond to attacks at an early stage, focusing on determining the most effective parameters for attack prevention. It will not only help reduce resource consumption but also improve the overall performance of the intrusion detection system. Furthermore, we aim to evaluate the effectiveness of this proposed model in a real vehicular environment to ensure its practical application.

Chapter 6

PREVIR: DDoS prevention framework

Attack prevention refers to implementing measures to stop an attack from happening or reduce its likelihood of success. In the context of VANET security, prevention measures aim to prevent unauthorized access, DDoS, and other attacks. Attack prevention is crucial as it helps minimize the risk of network attacks. VANET attacks are more sophisticated and devastating, ranging from financial loss to loss of life in some instances. Attack prevention is, therefore, a critical aspect of VANET security, as it helps to protect critical assets, including data, vehicles, and networks. Prevention measures reduce the likelihood of an attack by identifying vulnerabilities and implementing security measures to mitigate them before an attack occurs. These measures include access control, authentication, encryption, and intrusion detection systems. We implement these measures and safeguard VANETs from unauthorized access, data theft, and other types of attacks. Additionally, prevention is more cost-effective than remediation, which involves responding to an attack after attack occurrence. Remediation can be expensive and time-consuming and may only sometimes successfully restore the affected systems and networks to their previous state. Prevention, on the other hand, involves less cost and effort and is a more effective way of protection. Overall, attack prevention is essential in VANETs, as it minimizes the impact of cyber-attacks and protects critical assets from damage or loss.

6.1 Introduction

In this chapter, we describe ”Predictive Risk Evaluation for Vehicular Infrastructure Resilience (PREVIR),” which shows a pioneering use of a collaborative logit model and machine learning approaches for the prevention of DDoS assaults. PREVIR is a blended solution that uses statistics and machine learning approaches to develop an advanced DDoS attack protection solution. PREVIR offers several benefits by combining both strategies, including increased model efficiency and increased predictive power. By employing the

Wald test for parameter selection and model creation, as well as goodness-of-fit tests to assess the model's performance PREVIR makes use of a logit model and increases the effectiveness of current logit-based models. Additionally, the system uses DecisionStump as the base learner and the LogitBoost machine learning algorithm for packet categorization. The identification of malicious packets is made possible by statistical techniques that anticipate packet probability. Through frequent updates based on fresh data and iterative model refinement, the system constantly improves its performance.

The DDoS prevention capabilities offered by PREVIR show the utility of a hybrid strategy that combines statistics and machine learning techniques. Two datasets are used for experimentation: the NSL-KDD public dataset and a simulated and synthesized dataset created by us. These datasets cover a range of assaults, including R2L, U2R, TCP, and UDP flood, mixed flooding, and probe attacks. We show through a series of studies that PREVIR can classify packets with an accuracy of up to 99.99%. The model also obtains an average True Positive Rate (TPR) of up to 100%. PREVIR surpasses cutting-edge models, according to comparative studies, showing an average 20% increase in the protection of malicious packets.

6.2 Basic terminology: Logit model

The Cumulative Distribution Function (CDF) of random variable X is a probability that assumes values lower than or equivalent to X 0, where it is some observed exogenous value of X. We can formulate it as follows.

$$f(X = X_0) = p(X \leq X_0), \tag{16}$$

and the dependent variable is conditional, that is

$$P_i = E(Y_i = 1/X_i). \tag{17}$$

This implies the attack is malicious. Such non-linear probability function becomes as:

$$P_i = \frac{1}{1 + e^{-(\beta_1 + \beta_2 X_1 + \beta_3 X_2 ..... + \beta_6 X_5)}}, \tag{18}$$

thereby,

$$P_i = \frac{1}{1 + e^{-zi}} = \frac{e^{zi}}{e^{zi} + 1}. \tag{19}$$

This makes the following equation.

$$Z_i = \beta_1 + \beta_2(Flowpackets) + ... + \beta_n(Flowduration). \tag{20}$$

In Equation 20, $Flowpackets$, $Flowduration$ are the packet features. We can have n number of packet features. β is the co-efficient for a corresponding feature. The equation makes it clear that Zi ranges between -∞ and ∞, P_i ranges between 0 and 1 as the probability of malicious attack and legitimate traffic = 1. Therefore,

$$(P_i) + (1 - P_i) = 1. \tag{21}$$

In Equation 21,

$$(1 - P_i) = 1 - \frac{e^z}{1 + e^z}, \tag{22}$$

and

$$(P_i = \frac{e^z}{1 + e^z}). \tag{23}$$

$$(1 - P_i) = \frac{1}{1 + e^{zi}}. \tag{24}$$

Linear transformation becomes as:

$$\frac{P_i}{(1 - P_i)} = \frac{1 + e^{-zi}}{1 + e^{zi}} = e^{zi}. \tag{25}$$

Odd ratio is considered in favor of a malicious attack. We then log to both sides and we obtain:

$$Li = In(\frac{P_i}{(1-P_i)}) = Zi = (\beta_1 + \beta_2 X_1 + ... + \beta_6 X_5). \tag{26}$$

Parameter estimation In logistic regression, we should minimize log loss to obtain 'good fit' parametric values, which indicate the closeness of predicted probabilities to the corresponding actual values.

$$LogLoss = \frac{-1}{(N)} \sum_{i=1}^{n} Yi.Lu(Pi) + (1-Yi)In(i-Pi). \tag{27}$$

As an alternative method, the logit model uses maximum likelihood estimation to compute the log of odds ratio as shown in Equation 26.

6.3 Proposed model: PREVIR

We use the NS3 tool to generate three attack scenarios. One scenario contains benign traffic, the second has only attack traffic, and the third scenario includes benign traffic along attack traffic. During the simulation process, we use a random mobility model to generate a realistic environment with uncertain behavior of the vehicles. We capture the total traffic in a .pcap file. We observe this file in Wireshark and create a dataset in the .csv format. The attack scenario is based on a peer-to-peer-based reflection DDoS attack. The generated/ synthesized dataset contains all characteristics applicable in any generalized network scenario; however, the generation of the dataset uses a VANET architecture in NS3 with RSUs and vehicles (nodes). We apply the proposed system PREVIR to this synthesized and public dataset to check PREVIR's efficacy. The model selects only those attributes which may give the highest accuracy. PREVIR tests prevention efficacy in a vehicular environment. In the future, we will test PREVIR in realistic VANET.

In Galician, PREVIR denotes prevention. The rationale driving the PREVIR approach is to protect network infrastructure and services from DDoS attacks by developing a sophisticated and data-driven system for attack prevention. PREVIR predicts the probability of occurrence of malicious attacks through a logit model based on a simulated and public dataset to develop an early threat prevention system. The logit model offers both classification and computations of probabilities. We get the best outcomes using its mathematical capabilities with little training and without over-fitting. The learned weights (predicted parameters) provide information on the different variables' weight distribution. Additionally, it indicates if the connection is optimistic or adverse. We use an integrated logit model to analyze the correlation among these variables. It is a highly adaptable model that uses regularization, which lowers model error using regularization parameters. In this section, we discuss the overall functioning of PREVIR and the detailed process of model selection, dataset, parameter selection, goodness of fit, and probability calculation. We show the complete process in a flowchart through Figure 20.

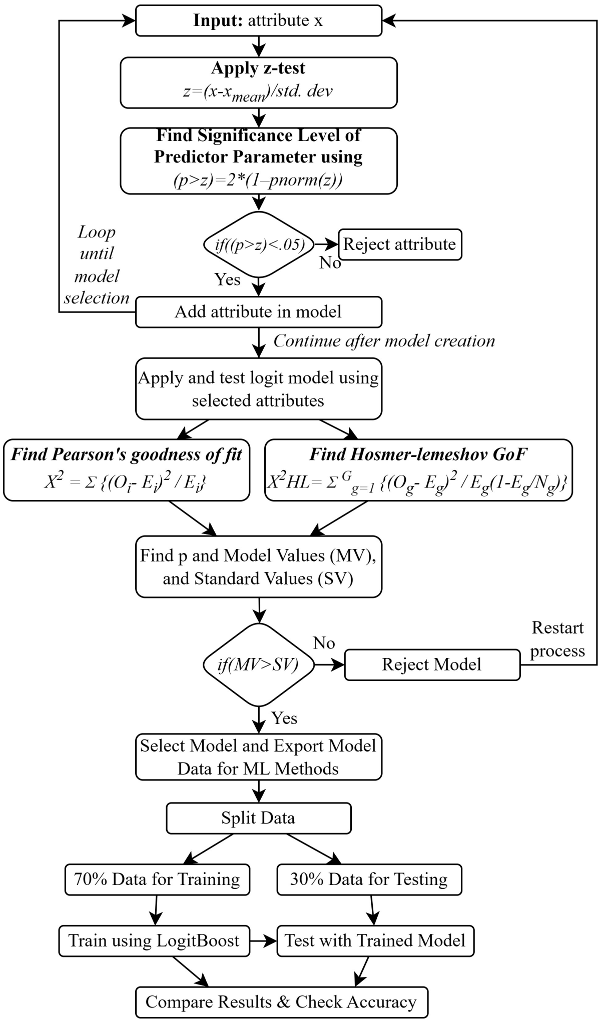
Input: attribute x
Apply z-test
$z=(x-x_{mean})/std.\ dev$
Find Significance Level of Predictor Parameter using
$(p>z)=2*(1-pnorm(z))$
$if((p>z)<.05)$
No
Reject attribute
Yes
Loop until model selection
Add attribute in model
Continue after model creation
Apply and test logit model using selected attributes
Find Pearson's goodness of fit
$X^2 = \Sigma\{(O_i- E_i)^2 / E_i\}$
Find Hosmer-lemeshov GoF
$X^2HL= \Sigma^G_{g=1} \{(O_g- E_g)^2 / E_g(1-E_g/N_g)\}$
Find p and Model Values (MV), and Standard Values (SV)
$if(MV>SV)$
No
Reject Model
Restart process
Yes
Select Model and Export Model Data for ML Methods
Split Data
70% Data for Training
30% Data for Testing
Train using LogitBoost
Test with Trained Model
Compare Results & Check Accuracy

Figure 20: Flowchart of PREVIR

6.3.1 Functionality overview

The proposed solution, PREVIR, is designed for prediction and classification tasks with a focus on feature selection and model evaluation. It begins by taking an attribute ’x’ as input and aims to produce predictions and classifications as output. The algorithm operates in a loop, repeatedly assessing each attribute in a given dataset. For each attribute, it performs a Z-test, standardizing the attribute’s values and calculating the significance level ’p’ for the predictor attribute. If the significance level is less than 0.05, the attribute is selected; otherwise, it’s rejected. This step helps in feature selection, identifying attributes that have a meaningful impact on the model.

Once the model’s attribute set is selected, a logistic regression model is applied using these chosen attributes. Subsequently, the algorithm computes the goodness of fit metrics, including the Chi-squared statistic and Hosmer-Lemeshow’s Goodness of Fit (GoF), to evaluate the model’s overall efficiency. By comparing the metrics, PREVIR decides whether to select or reject the model based on the criteria of Model Variance (MV) and Model Stability (SV).

After selecting the model, the algorithm proceeds to export the model data for machine learning applications. It then divides the data into training (70%) and testing (30%) sets. LogitBoost with Decision Stump initialization is employed for model training using the training features. In the testing phase, the model is applied to predict the output labels for observations in the testing dataset, storing the predicted labels for evaluation.

Finally, the algorithm evaluates the model’s performance using various metrics, such as accuracy and precision, to assess its effectiveness in predicting and classifying data. In essence, PREVIR combines feature selection, model evaluation, and machine learning techniques to enhance predictive accuracy and classification performance, making it a comprehensive approach for data analysis and prediction tasks.

PREVIR uses log functions to transform the probability functions. We change Non-linear equations (Equation 4) to linear equations (Equation 12). Equation 12 gives the probability in log odds form, where log odds depict the chances in favor of the outcome. Hence, it simplifies the task as coefficients such as b_1, b_2, b_3, and b_4 are linear in log odds. We apply

Wald's Z-test and individual probabilities using these coefficients. We also apply Hosmer-lemeshov's Goodness of fit to check the overall efficiency of the model. In computing results, we use p-values to see the effectiveness where 1.00 signifies a perfect fit model. We check individual packet probabilities using the coefficients to differentiate between malicious and benign packets. As we use logic and probit model with linear regression to prevent attacks in VANETs, we call our solution as *Predictive Risk Evaluation for Vehicular Infrastructure Resilience (PREVIR).*

6.3.2 Dataset

The three datasets that we used are listed below. We find different DDoS attack types in such datasets. If the proposed system or classifier successfully recognizes these attacks, we may use real vehicular networks to test and implement them. PREVIR is a statistical model that works accurately if appropriate data is available. We implement PREVIR on three datasets; two are pre-built (publicly available), and another is a synthesized dataset.

Generated dataset It is our own synthesized dataset that we made through simulation in NS3. We developed three different simulation scenarios to collect packets. In the first scenario, there is only benign traffic; in the second, there is only attack traffic; and in the third scenario, there are both benign and attack traffic. This scenario directs traffic from all intermediate nodes toward the victim, making it a deadly assault scenario. In NS3, we record all packet movement data in a .pcap file. We capture a total of 27 parameters with 697792 records. We also use an additional attribute for classification. We generate the dataset using the VANET characteristics shown in Table 22.

Public NSL-KDD dataset The KDD99 dataset is an improved dataset produced by the NSL-KDD. The University of New Brunswick developed this NSL-KDD dataset to fix several problems with the KDD99 dataset. A few instances of the assault types of DoS, Probe, U2R, and R2L, with their respective eleven, six, seven, and fifteen assault sub-classes, are included in this compilation. They enhanced NSL-KDD by removing redundancy and precisely splitting data for training and testing. Therefore, testing this dataset on a vehicular

Table 22: Attributes of Topology

Parameter	Value
Simulation Platform	NS3.2.7
No. of Vehicles	12
Attacker Nodes	1
Bot Nodes	10
Victim	1
No. of RSUs	1
Routing Protocol	UDP, TCP, ICMP
Visualisation Tool	NetAnim
DDoS Rate	20480 kbps
Normal data rate	512 kbps
Maximum Bulk Bytes	100000 kbps
Simulation Time	40 Seconds
Data Transmission Rate	100 Mbps
Communication Range	100 m X 100 m
Mobility Model	Random mobility

network proves to be feasible.

CIC-DDoS dataset The CIC-DDoS dataset, specifically the CIC-DDoS-2019, is typical actual network traffic, including benign and frequent DDoS assaults. The collection consists of labeled flows in CSV files and PCAP files that record network traffic data. Various parameters, including timestamps, source, and destination IP addresses, source and destination ports, protocols, and attack kinds, are used to categorize the flows. The researchers prioritized producing background traffic that simulates human interactions to provide a realistic environment for the dataset. To profile the abstract behavior of interpersonal encounters, they used the B-Profile approach, which was put out in a paper by Sharafaldin et al. in 2016. This system simulates the behavior of 25 people over various protocols.

6.3.3 Parameter selection

For model selection, we identify variables and check which can likely show an impact on the model. For example, in the first experiment with the generated dataset, we select five variables: flow packets, flow duration, total forward packets, total backward packets, and forward header length.

We apply Wald's Z-test to see which variables are important and contribute to DDoS prevention. Wrong variable selection may lead to wrong model selection and lead to failure.

$$z = (x - x_{mean})/std.dev \tag{28}$$

We observe and analyze the individual values of variables obtained by the Z-test. Results show that all the variables are statistically significant at a 5% confidence level. For instance, interpreting the variable TotalFwdPackets, we say that holding other variables constant, if 'TotalFwdPackets' increases by one, the average Logit value goes up by 0.04. That is, the log of odds in favor of the attack goes up by 0.04. We interpret all the variables in the same way.

6.3.4 Model construction and testing

After attribute selection, we construct the logistic regression model using the selected attributes. The logit model is a statistical model that predicts binary or multi-class outcomes. We then create the logit model and test using the chosen characteristics. Logit and Probit models have pros and cons; in our proposed PREVIR, we decided on the Logit model. The logit model predicts the probabilities of happening or non-happening of an event. These probabilities assist in attack prevention. Another idea behind the selection of the Logit model is that it uses the Cumulative Distribution Function (CDF) of the logistic distribution, whereas the Probit model uses standard normal distribution. We can use single or multi-class classification depending upon the attack packets. We can adapt the model if attack packets contain multiple packet types. It uses a regularization parameter that we use to fine-tune the model to reduce model errors. The Logit and Probit models yield the same results. If the number of observations is high in the distribution tail and values of Probit estimates become comparable with logistic estimates, we multiply logistic estimates by 0.625-factor value.

6.3.5 Overall goodness of fit

We apply LR Chi (Likelihood - Ratio Statistics) to check the aggregate significance of the model over the no variables (only intercept model). The results show that a p-value for LR chi is less than 5%, showing a significant impact of the selected parameters compared to the intercept model with no variables. We implemented the chi-square goodness of fit test using Equation 29.

$$X^2 = \sum \frac{(O_i - E_i)^2}{E_i}, \tag{29}$$

where X^2 is chi-square value, O_i is the observed frequency, and E_i is the expected frequency. We apply Hosmer-lemeshov's Goodness of fit to measure the overall model significance. It assumes the null hypothesis of the model is a good fit, and a p-value of more than 5 percent significance level shows the good fit model. Computed results show that a p-value of 1.00 signifies a perfect fit model. We calculate the Hosmer-lemeshov Goodness of fit is computed using Equation 30.

$$X^2HL = \sum_{g=1}^{G} \frac{(O_g - E_g)^2}{E_g(1 - E_g/N_g)}, \tag{30}$$

where O$_g$ signifies the observed events, E_g signifies the expected events and N_g signifies the number of observations for the gth group. G is the number of groups and $\sum$ is the summation notation. The test statistic follows a chi-squared distribution with $G - 2$ degrees of freedom. The output returns a chi-square value (a Hosmer-Lemeshow chi-squared) and a p-value (e.g., $Pr > ChiSq$). Small p-values mean that the model is a poor fit.

6.3.6 Computing probabilities

We can also compute probabilities besides computing coefficients in the Logit model. We use the given values of the explanatory variables to calculate the probabilities of attack. We use the "predict" command in Stata to generate these probability values. We indicate values of probabilities using p values, and these values range between 0 and 1. Values equal to 0 or near 0 indicate that the packet is benign, and values equal to 1 or near 1 indicate the packet is an attack packet. The predict command determines the predicted probability (packet probabilities) for each observation in the dataset using a logit model. The logistic function, sometimes called the sigmoid function, is the foundation of the internal equation to determine these probabilities.

$$p = 1/(1 + exp(-z)) \tag{31}$$

where:

p represents the predicted probability (packet probability) for an observation.

z represents the linear combination of the predictors in the logistic regression model, weighted by their respective coefficients.

6.3.7 Export model data for ML application

The next step is to export the model data for use in a machine learning (ML) environment once the model is trained and ready for use. We save the essential data so that the machine learning application can easily use the same for the model training.

6.3.8 Training

We often split the available dataset into two subsets: a training set and a testing set, to evaluate the performance of the trained model. We prepare the model using the training set and assess its performance using the testing set. A 70:30 data split is used, with 70% going to training and 30% to testing. Once we divide the data into training and testing sets, the following step is to load the 70% training dataset. The accuracy achieved by various ML algorithms at different data splits is shown below in Table 23.

Table 23: Accuracy results at various data splits

Split Ratio	**LB**	**ABM1**	**MLP**	**NBM**	**Vote**	**IMC**
40-60	99.9967	99.3284	66.9534	66.2951	66.2951	66.2951
50-50	99.9974	99.3247	85.9213	66.3467	66.3467	66.3467
60-40	99.9979	99.3279	87.9839	66.3754	66.3754	66.3754
70-30	99.9981	99.3317	70.2443	66.3955	66.3955	66.3955
80-20	99.9979	99.3257	85.6698	66.3409	66.3409	66.3409

In Table 23, it is evident that the accuracy kept on rising till 70-30 split and after then started declining. MultilayerPerceptron is one exception in which the highest accuracy is maximum at 60-40 data split. As the majority(84%) of algorithms give the highest accuracy at 70-30 data split, we decided to keep it for other datasets. The trained data model finds links and patterns in the data that make predictions. A popular machine-learning technique for classification problems is LogitBoost. We use the DecisionStump classifier to initialize the LogitBoost algorithm in this stage. Simple decision tree algorithms like DecisionStump work as weak learners within boosting algorithms like LogitBoost. LogitBoost with DecisionStump works in the following manner.

1. Initialize the training set:
   - Let N be the number of training instances.
   - Let N be the number of training instances. Initialize the instance weights w_i for $i = 1$ to N such that $1/N$ is assigned to each instance initially.
2. For each iteration $t = 1$ to T, where T is the maximum number of iterations:
   - Train a decision stump:
     - Let $f_t(x)$ represent the prediction of the t^{th} decision stump.
     - Select a feature j and find the best-split point s that minimizes the weighted impurity.
     - The decision stump's prediction for an instance x is:
   - Calculate the weighted error (ε) of the decision stump by summing the weights of misclassified instances.
   - Compute the stump's weight (α) using the equation: $\alpha = 0.5 * ln((1 - \varepsilon)/\varepsilon)$
   - Update the instance weights using the equation: $w_i = w_i * exp(\alpha)$ if instance i is misclassified $w_i = w_i * exp(-\alpha)$ if instance i is correctly classified
3. Combine the decision stumps weighted by their respective α values to form the final ensemble model.
4. To make predictions for new instances:
   - Each decision stump predicts either class 0 or class 1.

- The final prediction is obtained by summing the predictions weighted by their α values and applying the logistic function (sigmoid) to the sum:
  $y = sigmoid(\Sigma(\alpha_j * h_j(x)))$ where $h_j(x)$ is the prediction of the j^{th} decision stump and α_j is its corresponding weight.

6.3.9 Testing

Each observation in the testing dataset undergoes exposure to the trained model in this stage. The model predicts each observation's output label (classification) or value (regression) based on the discovered patterns and correlations from the training process. The system saves the predicted output label once the prediction is completed for each observation in the testing dataset. These projected labels are compared with the actual labels in the testing dataset to evaluate the model's effectiveness.

6.3.10 Evaluation

It is crucial to consider various performance measures when assessing a trained machine learning model's performance. Metrics like accuracy, precision, recall, and F1-score are significant in this situation. Accuracy indicates how accurately the model predicts outcomes by computing the ratio of cases predicted correctly to all instances. Precision measures the percentage of accurately detected positive cases among all positive predictions to assess the quality of positive predictions by evaluating the ratio of genuine positive predictions to the total number of positive cases. Recall, often called sensitivity, evaluates how well a model can spot positive events. We show this complete process in Algorithm 2.

6.4 Results and discussion

The results obtained from the implementation and evaluation of PREVIR, our hybrid framework for preventing DDoS attacks by integrating statistical methods and machine learning, demonstrate its robustness and effectiveness. Through a series of experiments using both our own simulated and synthesized dataset, as well as the widely used NSL-KDD public dataset, we were able to assess the performance of PREVIR in classifying packets and preventing

Algorithm 2 Proposed algorithm: PREVIR

```
Input: Attribute $x$
Output: Prediction and classification
repeat
    for each $x \in \mathcal{D}ataset$ do
        $z = (x - x_{mean})/std.dev$ { Z-test on attribute}
        $(p > z) = 2 * (1 - pnorm(z))$ {Significance level of predictor attribute}
        if $((p > z) < 0.05)$ then
            select attribute
        else
            reject attribute
        end if
    end for
until Model is selected
Apply logit model using selected attribute set
$X^2 = \sum \frac{(O_i - E_i)^2}{E_i}$ { Goodness of Fit }
$X^2 HL = \sum_{g=1}^{G} \frac{(O_g - E_g)^2}{E_g(1 - E_g/N_g)}$ { Hosmer-lemeshov GoF}
compute $p$, $MV$ and $SV$
if $(MV > SV)$ then
    select model
else
    reject model
end if
Export Model Data for ML application
Split Data in 70-30 Ratio
Load 70% training dataset
Initialize LogitBoost with DecisionStump
Train model with training features
for each $Observation \in \mathcal{T}estingDataset$ do
    Predict the output label using the trained model
    Store the predicted label
end for
Evaluate Performance metrics: Accuracy, precision
```

malicious attacks. The obtained results showcase the exceptional capabilities of PREVIR, with packet classification accuracy reaching an impressive 99.99%. Additionally, the system achieves a true positive rate (TPR) of up to 100% while maintaining an average false positive rate (FPR) of 35%. Moreover, comparative analysis reveals that PREVIR surpasses state-of-the-art models, exhibiting an average improvement of 20% in preventing malicious packets. These compelling outcomes affirm the efficiency and reliability of PREVIR as an advanced solution for vehicular infrastructure resilience against DDoS attacks.

We pass the packet through the model with the selected variables to identify maliciousness for a new packet that we have not evaluated in the past. Suppose the probability estimated by the model is greater than the threshold. In that case, we say it is a case of a potential attack, and if the probability is less than the threshold, we declare it a benign packet. Ultimately, we validate the calculated probabilities against actual observations and obtain the results. These compelling findings unequivocally establish the efficiency, reliability, and superiority of PREVIR as an advanced and indispensable solution for ensuring the resilience of vehicular infrastructure against the ever-growing threat of DDoS attacks.

6.4.1 Results of Z-Test

We employ the Z-test in the earlier phase to determine whether the parameters are significant. Similarly, we use the Z-Test to construct the logit model based on a specific collection of parameters from both datasets. We test 235,364 records in the Generated dataset, 125973 in the NSL-KDD dataset, and 1048575 records in the CIC-DDoS 2019 dataset. These datasets are evaluated and only relevant features are selected for inclusion in the model.

Generated dataset: We chose four variables for the first test with the generated dataset, i.e., time-to-live, time-since-previous-frame, and time-since-first-frame. This test evaluates p, z, and $p > z$ values to suggest which parameters suit PREVIR. The values of $p > z$ are most important in this test. As all the $p > z$ are 0 for all 4 attributes, this attribute combination is selected for the model. We show other resulting parameters obtained from the generated dataset in Table 24.

Abbreviations used in Table 24 are the Time To Live for TTL, Time Since the Previous

Table 24: Results of Wald's Z-test on Generated dataset

Class/Feature	Co-eff	Std.Err.	Z	P >z	95% Conf	Interval
Time	-0.869	0.170	-5.110	0	-1.2	-0.535
TTL	-8.138	1.782	-4.570	0	-11.6	-4.644
TSPF	-900.144	202.140	-4.450	0	-1296.3	-503.950
TSFF	1.229	0.248	4.950	0	0.7	1.716

Frame for TSPF, and Time Since the First Frame for TSFF.

NSl-KDD Dataset: Similarly, Table 25 presents the results of the Z-test performed on the NSL-KDD dataset. The table shows that all the $p > z$ values are less than the 5% significance level. Consequently, we consider variables with these values appropriate for the parameter selection procedure. To find the ideal combination, we analyzed a variety of attribute combinations. Forty-two characteristics and 125,973 occurrences make up the NSL-KDD Dataset. We have chosen the following five parameters for the model: duration, number of file creations, hot, number of failed logins, and count. Table 25 contains the findings we acquired using the NSL-KDD dataset.

Table 25: Results of Wald's Z-test on NSL-KDD Dataset

Class/Feature	Co-eff	Std.Err.	Z	P >z	95% Conf	Interval
DUR	-0.000	3.33e-06	-32.270	0	-0.000	-0.000
NFC	0.240	0.045	5.340	0	0.152	0.329
Hot	-0.041	0.002	-15.350	0	-.046	-0.035
NFL	-0.529	0.133	-3.960	0	-0.791	-0.267
Count	-.018	0.000	-166.860	0	-0.018	-0.018

Table 25 uses the abbreviations for variables: DUR represents Duration, NFC represents the Number of files created, and NFL represents the Number of Failed Logins.

CIC-DDoS 2019 Dataset: The CIC-DDoS 2019 dataset comprises multiple subsets of this CIC-DDoS dataset. We employed the Syn Dataset in our research. It has 1048575 records and 88 characteristics with the class labels Syn and Benign. While building the model, we select 5 parameters having the most relevance. These parameters are appropriate for the suggested Logit Model since they all generate a value of 0 for $p > z$.

Table 26: Results of Wald's Z-test on CIC-DDoS 2019 Dataset

Class/Feature	Co-eff	Std.Err.	Z	P >z	95% Conf	Interval
Protocol	-.08020	.00371	-21.59	0	-.08748	-.07291
Flowbytes	2.37e-1	2.39e-1	9.95	0	1.90e-1	2.84e-1
Flow Packets	5.88e-1	2.84e-1	20.72	0	5.33e-1	6.44e-1
Flag Count	-4.4733	0.31192	-14.34	0	-5.08468	-3.86198
Flow Duration	-1.37e-1	7.40e-1	-18.57	0	-1.52e-1	-1.23e-1

In Table 26, class label C1 stands for protocol, C2 for flowbytess, C3 for flow packets, C4 for flag count, C5 for flow duration

6.4.2 Results of Pearson's goodness of fit

After observing variables, we analyze the overall goodness of fit to see whether our proposed model is working fine or not. In Table 27, $Prob >chi2$ is 0 in both datasets; therefore, we claim that the PREVIR model is significant.

Table 27: Results of goodness of fit

Parameter	Generated	NSl-KDD	CIC-DDoS 2019
Number of obs	253364	125973	1048575
LR chi2	46298.38	36.46	72.84
Prob >chi2	0.0000	0.0000	0.0000
Pseudo R2	0.992	0.295	0.412
Log likelihood	-17.942503	- 43.559954	- 28.54

6.4.3 Results of Hosmer-Lemeshov goodness of fit

Table 28 shows the results of probabilities and co-variate. This test produces $Prob >chi2$ as 1.000; therefore, we accept the null hypothesis is accepted. The tests for both datasets results show the probability value of 1.

Table 28: Results of Hosmer-Lemeshov goodness of fit

Parameter	Generated	NSL-KDD	CIC-DDoS 2019
Total observations	235364	125973	1048575
Pearson chi2	146.09	44693.63	78.45
Prob >chi2	1	1	1

6.4.4 Individual packet probabilities

PREVIR shows that the probability of ”attack packets” is 0.9 or more; it signifies that our model’s prediction for the attack is accurate. Therefore, we can claim that PREVIR is usable to prevent future attacks by using these probability values. Values near 0 or precisely 0 indicate a benign packet the system can process. We show the results in Table 29.

Table 29: Individual packet probabilities

No	Time	Source	Dest.	EpochTime	p
1	0	10.1.1.1	10.1.2.2	0.002009	0.999997
2	0	10.1.2.2	10.1.1.1	0.002009	0.989718
3	0.000277	10.0.0.1	10.1.2.2	0.002286	0.999997
4	0.000321	10.0.0.5	10.1.2.2	0.002330	0.999997
5	0.000364	10.0.0.9	10.1.2.2	0.002373	0.999997
6	0.000407	10.0.0.13	10.1.2.2	0.002416	0.999997
7	0.000451	10.0.0.17	10.1.2.2	0.002460	0.999997
8	0.000494	10.0.0.21	10.1.2.2	0.002503	0.999997
9	0.000537	10.0.0.25	10.1.2.2	0.002546	0.999997
10	0.000581	10.0.0.29	10.1.2.2	0.002590	0.999997

6.5 Classification results

6.5.1 Accuracy

In our study, we evaluated the performance of various machine learning algorithms, namely LogitBoost, AdaBoostM1, MultilayerPerceptron, NaiveBayesMultinomial, Vote, and InputMappedClassifier, in terms of their accuracy. The LogitBoost achieves the highest accuracy in the Generated Dataset with a perfect score of 99.99%. AdaBoostM1 follows closely with an accuracy of 99.33%. The remaining algorithms, including MultilayerPerceptron, NaiveBayesMultinomial, Vote, and InputMappedClassifier, have lower accuracy scores of 70.24%, indicating relatively less accurate predictions. The NSL-KDD Dataset’s accuracy score ranges from 49.01% to 83%. LogitBoost performs best with an accuracy of 83%, while AdaBoostM1 and MultilayerPerceptron achieve accuracy scores above 82%. The CIC-DDoS 2019 Dataset shows high accuracy for all algorithms, with MultilayerPerceptron, AdaBoostM1, and LogitBoost performing exceptionally well, with accuracy scores above 99%.

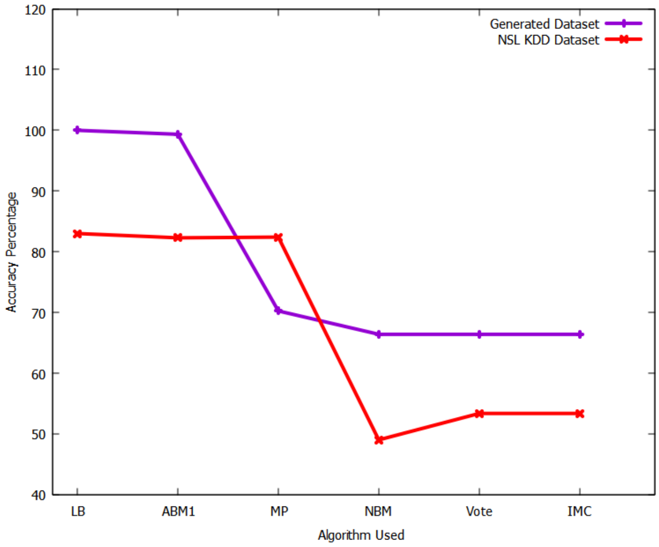

Figure 21: Accuracy of the algorithms

6.5.2 Model construction time

The model construction time varies across the datasets and algorithms. In the Generated Dataset, LogitBoost takes the longest time to construct, with a duration of 27.12 seconds. AdaBoostM1 and NaiveBayesMultinomial have shorter construction times of 12.36 and 0.06 seconds, respectively. On the other hand, MultilayerPerceptron exhibits the longest construction time of 252.41 seconds, indicating a more time-consuming process. In the NSL-KDD Dataset, the construction times are lower overall, with NaiveBayesMultinomial, Vote, and InputMappedClassifier having the shortest construction times of 0.01, 0.02, and 0.03 seconds, respectively. MultilayerPerceptron takes the longest time to construct among these algorithms, with a duration of 31.88 seconds. The construction times in the CIC-DDoS 2019 Dataset are relatively higher, with NaiveBayesMultinomial, Vote, and InputMappedClassifier exhibiting the shortest construction times. However, even the longest construction time in this dataset, observed for MultilayerPerceptron with 748.27 seconds, is considerably lower than the Generated Dataset.

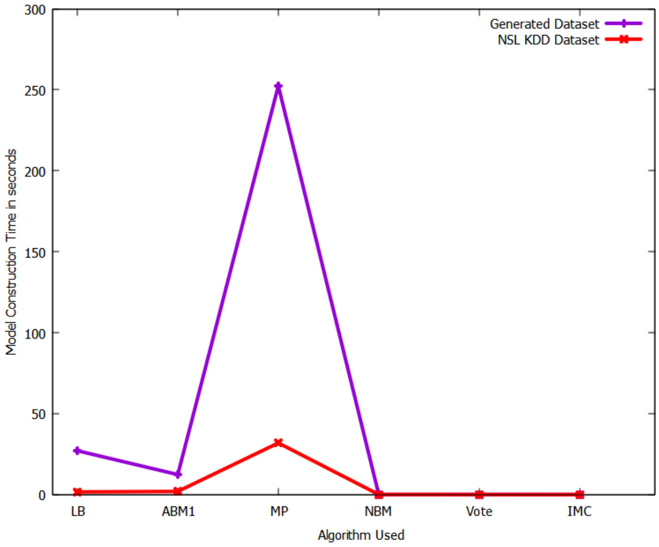

Figure 22: Model Construction Time

6.5.3 Testing time

AdaBoostM1 has the shortest testing time of 0.14 seconds in the Generated Dataset, indicating its prediction efficiency. Conversely, MultilayerPerceptron and Vote require more time for testing, with respective durations of 0.56 and 0.27 seconds. In the NSL-KDD Dataset, the testing times are relatively low across all algorithms, with MultilayerPerceptron, NaiveBayesMultinomial, and Vote having the shortest testing times of 0.03 seconds. The CIC-DDoS 2019 Dataset shows slightly higher testing times, with AdaBoostM1 having the shortest duration of 0.28 seconds, while InputMappedClassifier requires the longest time of 0.72 seconds.

6.5.4 TP & FP Rate

In our generated dataset, LogitBoost achieves a perfect TP Rate of 1 and an FP Rate of 0, indicating excellent performance in correctly identifying positive instances and avoiding false positives. AdaBoostM1 exhibits a high TP Rate of 0.99 and a low FP Rate of 0.01. MultilayerPerceptron, NaiveBayesMultinomial, Vote, and InputMappedClassifier have TP and FP Rates of 0.7 and 0.14, respectively. In the NSL-KDD public dataset, LogitBoost, AdaBoostM1, and MultilayerPerceptron have TP Rates of 0.83 and FP Rates of 0.17 or 0.18. NaiveBayesMultinomial, Vote, and InputMappedClassifier have TP and FP Rates of

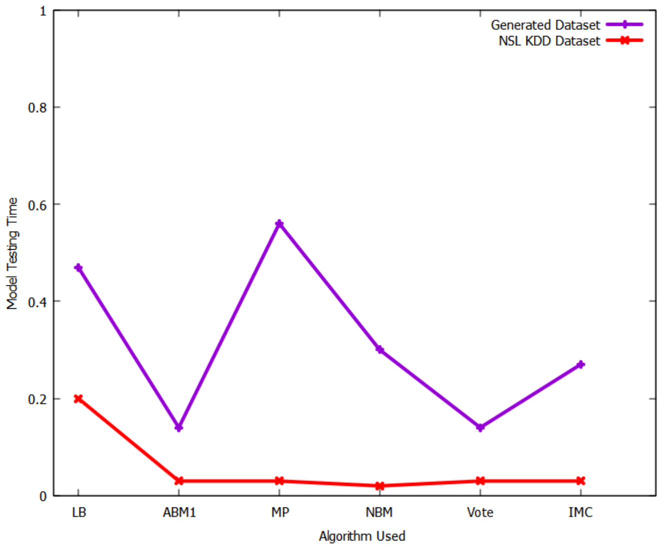

Figure 23: Model Testing Time

0.49 and 0.45 or 0.53, respectively. Similarly, in the CIC-DDoS 2019 Dataset, FP rates are consistently high for all algorithms, suggesting a relatively high misclassification rate of negative instances.

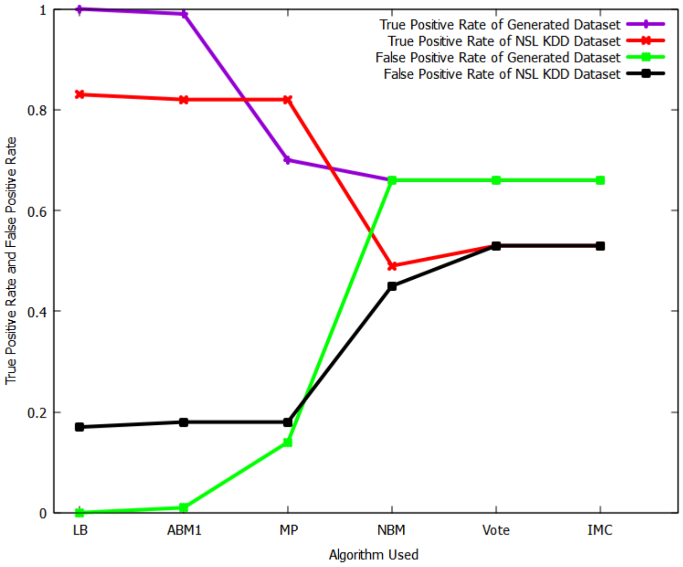

Figure 24: True Positive Rate and False Positive Rate

6.5.5 Precision, recall, and F1-score

In our generated dataset, LogitBoost achieves perfect precision, recall, and F1-score (all are equal to 1). However, the values for AdaBoostM1, MultilayerPerceptron, NaiveBayesMultinomial, Vote, and InputMappedClassifier are available, making it difficult to assess their performance in terms of precision, recall, and F1-score accurately. In the NSL-KDD public dataset, LogitBoost, AdaBoostM1, MultilayerPerceptron, and NaiveBayesMultinomial exhibit similar precision, recall, and F1 scores. However, we use the precision values for Vote and InputMappedClassifier provide must be provided, making it difficult to assess their precision, recall, and F1-scores accurately. In the CIC-DDoS 2019 Dataset, Multi-Layer Perceptron (MLP) achieves the highest precision and F1-score among the algorithms.

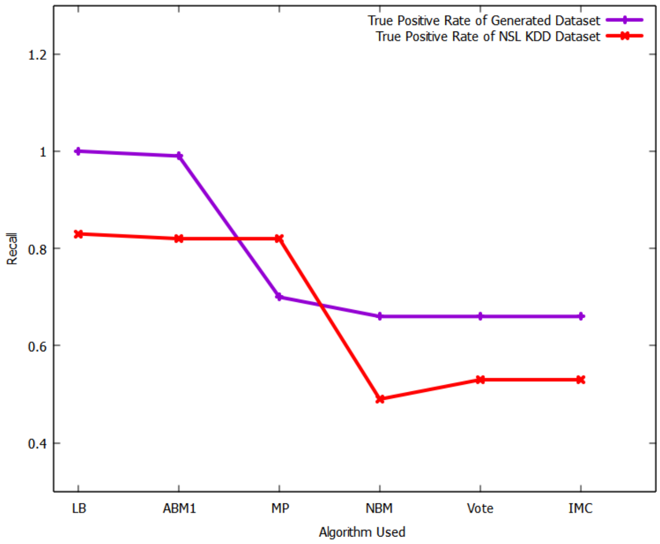

Figure 25: Recall values in both the datasets

The analysis of the datasets reveals variations in model construction time, testing time, accuracy, TP Rate, FP Rate, and, to some extent, precision, recall, and F1-score across different algorithms. While LogitBoost generally performs well in terms of accuracy and TP Rate, the performance of other algorithms may vary depending on the dataset. It is worth noting that the missing values for precision, recall, and F1-score limit a comprehensive analysis of these metrics for some algorithms in both datasets.

6.6 Comparative analysis

We compare the results of our proposed PREVIR with the existing state-of-the-art models for validation. We consider two existing models as mentioned in Green et al. [241] and B. R. Bajracharya [252] as these two models are closely connected to PREVIR in the use probabilistic approach. We show the statistical results in Figure 26. In our experiments, we use two tests, one with our synthesized dataset and another with the publicly available dataset. In this work, we have considered the mean values of both tests.

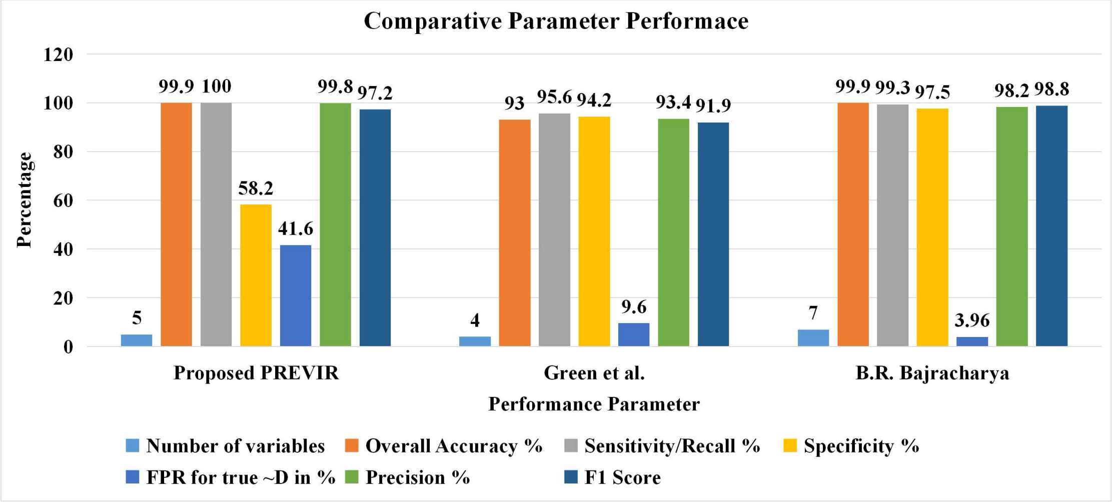

Figure 26: Resulting parameters of comparative studies

Comparing the classification results of PREVIR in the generated dataset and the NSL-KDD dataset, we observe significant variations in model construction time, testing time, accuracy, TP Rate, FP Rate, and, to some extent, precision, recall, and F1-score. In the generated dataset, PREVIR's LogitBoost achieves the highest accuracy of 99.99% and exhibits excellent TP and FP rates performance. AdaBoostM1 also performs well with a high TP Rate of 0.99. However, some algorithms limit precise precision, recall, and F1-score analysis due to missing values. On the other hand, in the NSL-KDD dataset, the accuracies range from 49.01% to 83%, with LogitBoost and AdaBoostM1 achieving the highest accuracies. The TP Rates and FP Rates show variations among algorithms, with some achieving better performance than others. Again, the availability of precision, recall, and F1-score values is limited, hindering a comprehensive comparative analysis. The generated dataset demonstrates higher overall accuracy and better TP Rate and FP Rate performance than the NSL-KDD dataset. However, the analysis must include both datasets' precision, recall, and F1-score values.

In addition to this comparative analysis, we compare other DDoS prevention solutions as shown in Table 30. In this table, we compare PREVIR with four existing solutions. These solutions include Bloom filters with IP-CHOCK, genetic model, reCAPTCHA controller, and HCPDS-based framework. We compare the techniques based on features and dataset, simulator type, simulation timing, area of communication and protocol, advantages and limitations, network topology, and performance accuracy. From this comparative study, we claim that our proposed PREVIR is significant for DDoS detection and obtains good accuracy.

Table 30: Comparison of Various strategies with proposed model

Technique	**Bloom filters with IP-CHOCK**	**Genetic Model**	**re CAPTCHA controller**	**HCPDS based framework**	**PREVIR**
Reference	Verma et. al. (2013) [247]	Malhi et. al. (2016) [248]	Poongodi et. al. (2019)[249]	Prabakeran et. al. (2020) [250]	Proposed approach
Year	2013	2016	2019	2020	2022
Feature Set	Source IP, Destination IP, ACK/SYN	speed, direction and location of the vehicle, feedback rate, exponential backoff and associativity time	Source IP, Destination IP, Source Port, Destination Port	(packet factors, RSU zone, and vehicle dynamics)	flow packets, flowduration, totalfwdpackets, totalbackwardpackets
Dataset	Simulation based dataset	Simulation based dataset	Simulation based dataset	Simulation based dataset	Simulation based dataset
Advantages	Support Filtering, Prevents small and Large attacks	High accuracy, low tracking time of attackers and low network recovery time.	High efficiency, low latency & energy consumption	Suitable for security and privacy preservation	High accuracy, sensitivity and specificity
Limitation	Low acceptance rate during attack	Exact results are not provided	Memory overheads	Less accurate and slow	Not such problems
Comparison based on simulation and network topology					

Simulator	NS-2.34, NAM, tracing	MATLAB	NS-2.28	NS2	NS-3
Simulation time	80 s	100 sec	20s	100 sec	40 Sec
Communication area	200m	500m	1500 X 1500 m	1000 X 1000 m	100 X 100 m
Packet size	Not Provided	Not Provided	Not Provided	Not Provided	Default
Number of Nodes	25, 50, 75, 100, 125, and 150	50–400	50, 100, 150, and 200	50 to 400	12
Network type	DSRC with Two Ray Ground Model	Not Provided	DSRC with Two Ray Ground Model	DSRC	DSRC
Mac Type	IEEE 802.11p	IEEE 802.11p	IEEE 802.11p	IEEE 802.11	IEEE 802.11
Data transmission speed	5, 10, 15, 20, and 25 m/s	Not Provided	10kbps	Not Provided	512kbps, 20480kbps
Comparison based on Results Obtained					
Accuracy	83.30%	89.9%	94.7%	99.6%	100%

6.7 Conclusion and future Work

In the presented work, we show PREVIR as a novel composite model of machine learning and statistical analysis for DDoS detection in vehicular networks. With its probability-based approach and tailored defense mechanisms, PREVIR enhances preventive capabilities, adapts to real-time data, and provides robust protection against various DDoS attacks. By addressing the research gap in VANET security and contributing to the development of dedicated prevention systems, PREVIR is a valuable solution for mitigating DDoS attacks and ensuring the resilience of VANET infrastructures.

In our study, LogitBoost consistently demonstrated the highest accuracy values across different datasets, achieving a perfect score of 99.99% in the Generated Dataset and an accuracy of 83% in the NSL-KDD Dataset. In the CIC-DDoS 2019 Dataset, LogitBoost, AdaBoostM1, and MultilayerPerceptron performed exceptionally well, with accuracy scores above 99%.

These results provide empirical evidence of LogitBoost's superior performance in accurately predicting outcomes compared to other evaluated algorithms. Similarly, the evaluation of machine learning algorithms on multiple datasets reveals that LogitBoost consistently performs excellently in correctly identifying positive instances and avoiding false positives, as evidenced by its perfect TP rate of 1 and FP rate of 0 in the generated dataset. AdaBoostM1 also exhibits strong performance with a high TP rate of 0.99 and a low FP rate of 0.01. However, other algorithms such as MultilayerPerceptron, NaiveBayesMultinomial, Vote, and InputMappedClassifier show lower TP and FP rates.

In our analysis, LogitBoost achieved perfect precision, recall, and an F1-score of 1 in the generated dataset, indicating its exceptional performance. In the NSL-KDD public dataset, LogitBoost, AdaBoostM1, MultilayerPerceptron, and NaiveBayesMultinomial exhibited similar precision, recall, and F1 scores. Among the algorithms for the specified metric, the Multi-Layer Perceptron (MLP) algorithm achieved the highest precision and F1 score in the CIC-DDoS 2019 Dataset. We also compare our proposed PREVIR with non-amalgamated and non-ML contemporary solutions for DDoS prevention. Our proposed PREVIR outperforms all the frameworks and shows 100% accuracy of probability analysis and classification.

In future research, it is essential to focus on improving the performance metrics of the PREVIR model by addressing missing values in precision, recall, and F1-score evaluation. PREVIR should also be evaluated on diverse datasets to assess its performance in different network environments. Further efforts should be directed towards implementing PREVIR in real-time DDoS detection systems, considering scalability and latency issues. Lastly, mitigating the false positive rate of the model is vital to minimize disruptions to legitimate network traffic and improve its usability in practical scenarios.

Chapter 7

Conclusion and Future Work

7.1 General

In this conclusion chapter, we summarize the findings and contributions of the thesis and present its limitations and future research directions. This thesis focuses on the problem of Distributed Denial of Service (DDoS) attacks in Vehicular Ad-hoc Networks (VANETs) and the proposed solution for detecting and preventing such attacks. The problem of DDoS attacks in VANETs is becoming increasingly critical as more and more vehicles are being equipped with communication systems, making them vulnerable to such attacks.

7.2 Summary of Findings

The proposed solution is a combination of machine learning algorithms and statistical methods, which makes it a robust solution that can adapt to the changing dynamics of VANETs. The simulation results show that the proposed solution can detect DDoS attacks with high accuracy and prevent them from affecting the network performance. The results also show that the solution has low false positive and false negative rates, indicating its effectiveness in detecting real DDoS attacks and avoiding false alarms. Additionally, the solution is designed to be scalable, making it suitable for large-scale VANETs.

7.2.1 Results of VAIDANSHH

The results obtained from VAIDANSHH's performance evaluations highlight its effectiveness in enhancing the security of VANETs. VAIDANSHH's layered approach to intrusion detection, which encompasses monitoring hardware resource consumption, flow parameters, and application-level attacks, demonstrates a holistic and comprehensive strategy for safeguarding VANET communication channels.

One of the notable findings is VAIDANSHH's adaptability to the dynamic nature of VANETs. By dynamically adjusting traffic thresholds and employing minimal packet features, the system excels in identifying both known and previously unseen attack patterns. This adaptability is crucial in real-world scenarios where attackers continuously evolve their tactics.

Additionally, the deployment strategy of VAIDANSHH at key points in the VANET infrastructure, such as RSUs and TAs, provides a strategic advantage. By monitoring traffic at these critical junctures, VAIDANSHH contributes to early threat detection and rapid response, ultimately enhancing the safety of drivers and passengers.

The low rate of false positives achieved by VAIDANSHH is another significant result. This indicates that the system effectively distinguishes between normal network behavior and potentially malicious activity, reducing unnecessary alerts and ensuring that security personnel can focus their attention on genuine threats. This is a critical aspect of any intrusion detection system, as excessive false positives can lead to alert fatigue and diminished effectiveness.

This thesis provides a complete parameter-based DDoS detection solution that uses the AAM to spot abnormal network activity. The suggested model uses a variety of variables, including jitter, latency, and packet loss ratio, to efficiently detect assaults with high accuracy, precision, recall, and F1-Score. Under the suggested model, the BayesNet method works remarkably well, with a weighted TPR of 1 and a weighted false-positive rate of 0. The VAIDANSHH model may be applied to various factors and is effective at almost 100% accuracy in identifying attacks. The model's synthesized dataset accurately depicts vehicular networks and is realistic. Other researchers working on intrusion detection in VANETs can use the suggested solution to comprehend the nature of DDoS attacks. The thesis makes a significant addition to the topic of DDoS detection overall and emphasizes the value of flow metrics in spotting suspicious activity and averting attacks.

7.2.2 Results of PREVIR

PREVIR's results underscore its prowess as a DDoS attack prevention system tailored for VANETs. The transformation of non-linear probability functions into linear forms using logarithmic functions simplifies the modeling process while maintaining effectiveness in a dynamic VANET environment.

The results of PREVIR's performance evaluations are particularly striking. Achieving 100% accuracy in both test scenarios and 99.99% accuracy in test-2 demonstrates PREVIR's exceptional capability to accurately detect and prevent DDoS attacks. This high level of accuracy is essential in ensuring the continued availability and integrity of VANET communication.

The emphasis on low false positive rates in PREVIR's results is noteworthy. Despite the high accuracy in attack detection, the system maintains false positives at a minimum, with only a 9.6% false positive rate for real DDoS attacks. This result showcases PREVIR's precision and reliability, as it avoids unnecessary disruptions to legitimate network traffic.

Furthermore, PREVIR's comparative analysis with other DDoS prevention technologies highlights its strengths and advantages. Its effectiveness, scalability, and adaptability make it a standout choice for protecting VANETs against DDoS attacks.

Similarly, PREVIR, works by transforming the equation for the non-linear probability function into a linear form using the log function. PREVIR assesses how well it performs using p-values and Wald's Z-test. PREVIR simulates vehicle communication in VANET with an attack-based topology and performs effectively when adequate data is available. The system uses NS3, develops a dataset through simulations, and identifies relevant variables using the Z-test and parameter selection. The Z-test chooses a set of variables for the PREVIR system that can provide a superior model. The Z-test works on two different variable sets from two datasets to establish the overall goodness of fit and Hosmer-Lemeshov's goodness of fit values. According to the results from the first dataset, four variables—time, time-to-live, time-since-previous-frame—are adequate for the proposed PREVIR model. We also evaluate performance metrics like sensitivity, specificity, accuracy, TPR, and FPR. We verify the PREVIR model's efficacy in preventing DDoS attacks by comparing its results with two other state-of-the-art models, attaining 100% accuracy in both tests and 99.99% in test-2. Despite being 100%, the specificity only averages 16.67%, making it 58.33% specific. For real D, the false positive rate is 9.6%. The F1 score is displayed, and the precision percentages are 98.59% and 97.92%, respectively.

In summary, the results obtained from VAIDANSHH and PREVIR's evaluations provide compelling evidence of their effectiveness in bolstering the security and resilience of VANETs. Their adaptability, low false positive rates, and ability to achieve high accuracy in attack detection make them invaluable assets in safeguarding the integrity of VANET communication channels and ultimately ensuring the safety of drivers and passengers. These results underscore the significance of these solutions in addressing the critical security challenges faced by VANETs in the modern era.

Finally, we compare PREVIR with four other DDoS prevention technologies to evaluate their features, datasets, simulator type, timing, communication area and protocol, advantages and disadvantages, network architecture, and performance accuracy. PREVIR is a good choice and performs well in accuracy.

7.3 Novelty and Innovation

VAIDANSHH and PREVIR stand out as innovative solutions in the realm of VANET security due to their unique approaches and exceptional performance. VAIDANSHH's multi-layered intrusion detection approach, dynamic thresholding, and strategic deployment at critical network points set it apart from traditional single-layer solutions. Its ability to maintain a low false positive rate while adapting to dynamic VANET conditions showcases its innovation. PREVIR, on the other hand, introduces log transformations for scalability, rigorous statistical model evaluation, and insightful comparative analysis. Its near 100% accuracy in DDoS attack prevention is substantiated by comprehensive testing methodologies and realistic datasets, making it a pioneering solution in safeguarding vehicular networks. These innovations collectively address the complex challenges of VANET security and significantly enhance the resilience of vehicular infrastructure.

While the reported results indicate the potential of VAIDANSHH and PREVIR, it's important to approach real-world implementation with the understanding that actual results may vary. Rigorous testing in diverse scenarios is a positive indicator of their capabilities, but the dynamic and unpredictable nature of VANETs means that continuous monitoring, adaptation, and ongoing evaluation are essential to ensure the effectiveness of these solutions in practical deployments. Here are some assumptions or considerations on which our results are based:

Network Environment: During testing, there may be assumptions about the stability and characteristics of the network environment. For instance, it might be assumed that the network infrastructure is well-maintained and that communication links are reliable. Real-world networks, however, can be subject to various issues, such as hardware failures, network congestion, or interference, which may not align with the assumptions made during testing.

Traffic Patterns: Assumptions are often made about traffic patterns, including the volume and types of data being exchanged among vehicles and infrastructure. Testing may use synthetic or controlled traffic patterns that may not fully represent the diversity and unpredictability of real-world traffic. Actual traffic patterns can vary significantly, affecting the behavior of security mechanisms.

Attack Behaviors: Assumptions about the behavior and strategies of attackers can influence testing scenarios. Testing typically involves known attack patterns or simulated attack behaviors. However, real-world attackers are adaptive and may employ novel tactics that were not considered in testing. Additionally, the intensity and frequency of attacks can fluctuate over time.

Impact on Accuracy: The accuracy of security frameworks is highly dependent on the accuracy of these assumptions. If the assumptions made during testing do not align with the real-world conditions, the accuracy of the framework may vary in practice. Here's how this misalignment can impact accuracy:

False Positives/Negatives: Inaccurate assumptions can lead to false positives (incorrectly identifying normal traffic as an attack) or false negatives (failing to detect actual attacks). For example, if testing assumes a low level of network congestion, but congestion is high in the real world, it may lead to false alarms or missed attacks.

Effectiveness: Assumptions about attack behaviors can impact the framework's effectiveness in detecting new or evolving threats. If the framework's assumptions do not encompass these threats, it may not perform well against them.

Robustness: The robustness of the framework, or its ability to adapt to changing conditions, can be compromised if assumptions about network stability or traffic patterns do not hold. Real-world fluctuations may challenge the framework's ability to maintain high accuracy.

7.4 Contribution to the Field

This thesis makes a significant contribution to the field of VANET security, by providing a practical and effective solution for detecting and preventing DDoS attacks. The proposed solution provides a novel combination of machine learning algorithms and statics-based methods, which makes it a robust solution that adapts to the changing dynamics of VANETs. Furthermore, we designed the solution to be scalable, making it suitable for large-scale VANETs.

VAIDANSHH and PREVIR are two innovative DDoS attack detection and prevention systems that have made significant contributions to society in network security. VAIDANSHH, a network intrusion detection system, provides a layered approach to protecting the VANET communication channel by monitoring hardware resource consumption, flow parameters, and application-level attacks. Its adaptive traffic thresholds and minimal packet features for attack detection ensure high efficiency and effectiveness in detecting attacks while reducing false positives. By deploying VAIDANSHH at RSUs or TAs, we can monitor VANET traffic more securely, increasing safety for drivers and passengers. Similarly, PREVIR has also made a substantial contribution to society by providing an effective and accurate method for detecting and preventing Distributed Denial of Service (DDoS) attacks in VANETs. Its statistical model evaluates the performance of the Logit and Probit models using p-values and Wald's Z-test and identifies significant variables for effective parameter selection. The model achieved 100% accuracy in both tests and 99.99% accuracy in test-2, making it highly effective in thwarting DDoS attacks. By conducting a comparative analysis of features and datasets, VAIDANSHH and PREVIR have provided more advanced incident response methods and enhanced the overall security of VANET communication channels. These detection systems have thus made a significant contribution to society by improving the safety of drivers and passengers and protecting the integrity of VANET systems.

Table 31: Important characteristics of the proposed models in the research work

Proposed Model	**Salient features**	**Issues addressed**
VAIDANSHH	Three-tier architecture	It provides a safe vehicle environment by applying rigorous security checks.
	Adaptive IDS	It addresses the problem of reliable results by considering real-time traffic.
	Heterogeneity	It addresses the problem of incorporating vehicles of different vendors, standards, protocols, and technologies.
	Dataset	We provide a synthesized dataset and address the issue of testing on a reliable dataset.
	BayesNet ML algorithm	We tested the effectiveness of BayesNet in VANETs for the very first time.
PREVIR	Logit model-based	Logistic regression is a novel approach in the prevention of DDoS in vehicular networks.
	Compact feature set	PREVIR addresses the problem of DDoS prevention with minimal features.
	Results	It provides efficient results in terms of accuracy, sensitivity, TPR, and FPR.
	Rigorous validation	We validate our results by rigorous testing through Wald's Z-test, HL Goodness of Fit, and individual packet probabilities.

7.5 Limitations and Future Work

While the proposed solution is effective in detecting and preventing DDoS attacks in VANETs, there are still some limitations that need to be addressed in future work. For example, the solution does not consider other types of attacks and does not incorporate additional security measures to further improve the security of VANETs. Additionally, the simulation results are based on a specific set of assumptions and scenarios, and further testing is required to validate the solution in different scenarios.

One area that needs improvement is the standard datasets required for testing and comparing detection and prevention systems. The availability of standardized datasets would make it easier for academics to create and assess their models and algorithms on an equal playing field, enabling more precise comparisons of the efficacy of various systems.

Another area for improvement is the dynamic nature of DDoS attacks, which makes it difficult to design systems that can adapt and respond to new and emerging attack tactics.

Attackers frequently change their strategies, making staying on top of distinctive attack patterns challenging.

Additionally, real-world scenarios' huge network traffic volumes make identifying attacks difficult while keeping false favorable rates low. False positives cause disruptions and pointless alarms by mistaking legitimate traffic for harmful.

Finally, developing and deploying efficient detection and prevention systems is challenging due to the complexity of the communication network infrastructure. The resources available for detection and prevention may be limited in some circumstances, and the system must be compatible with various network topologies.

7.6 Future Research Directions

Future research focuses on enhancing the proposed solution by considering other types of attacks and incorporating additional security measures to further improve the security of VANETs. We can conduct additional testing to validate the solution in different scenarios and environments. One possible suggestion is to investigate the effectiveness of VAIDANSHH and PREVIR under different network topologies and attack scenarios. These network topologies and attack scenarios help improve these systems' generalizability and ensure their effectiveness in various real-world settings. Additionally, we can explore more advanced machine learning techniques, such as deep learning to improve the accuracy of intrusion detection and prevention. Finally, we can do more research to develop efficient and scalable approaches to manage the large amounts of data generated by VANETs to support the effective operation of VAIDANSHH and PREVIR. These future research directions have the potential to advance the field of VANET security and contribute to the development of more effective and reliable intrusion detection and prevention systems.

Another future research direction is to develop more advanced machine learning algorithms and deep learning models for DDoS detection and prevention. While the current state-of-the-art models, such as VAIDANSHH and PREVIR, have shown promising results, they still have limitations in handling sophisticated and distributed attacks. We can develop more advanced algorithms and models to improve the accuracy and speed of attack detection and the ability to adapt to new attack patterns. Another potential research direction is to investigate the use of blockchain technology in DDoS prevention. Blockchain technology is effective in preventing attacks on financial systems and data breaches. Implementing a blockchain-based solution for DDoS prevention may improve the security and resilience of VANETs against attacks. Finally, we can conduct research on the development of more efficient and secure communication protocols for VANETs. Current protocols such as DSRC and LTE-V are vulnerable to various attacks, and there is a need for more secure and efficient communication protocols that can protect against DDoS attacks and other types of threats. By developing more advanced communication protocols, it may be possible to create a more secure and reliable VANET infrastructure that can better support the needs of intelligent transportation systems.

7.7 Conclusion

This thesis presents a comprehensive study of DDoS attacks in VANETs and the proposed solution for detecting and preventing such attacks. The results of this study indicate that the proposed solution can effectively detect and prevent DDoS attacks in VANETs, thus enhancing the security of the network. The proposed solution is a significant contribution to the field of VANET security, and there are many opportunities for further research to enhance the solution and address its limitations.

In conclusion, our research has presented two effective approaches for detecting and preventing DDoS attacks in vehicular networks. VAIDANSHH, a Network Intrusion Detection System, uses adaptive traffic thresholds and minimal packet features to identify attacks at all three security tiers, resulting in a highly efficient and effective model. On the other hand, PREVIR is a statistical model that performs well when relevant data is available, using the log function to convert the equation for the non-linear probability function into a linear version. By comparing and contrasting these two approaches, we have identified the strengths and limitations of each method and the conditions under which they are most effective. While VAIDANSHH is suitable for deployment at an RSU or TA, PREVIR requires relevant data for performance. However, our research demonstrates that both models are promising for improving the security and reliability of vehicular networks in the face of DDoS attacks, thus contributing to the development of safer and more efficient transportation systems.

References

[1] Kizza Joseph Migga Guide to Computer Network Security [Book]. - University of Tennessee Chattanooga, TN, USA: Springer Nature, June 2017.

[2] Migues, S., & Housley, R. (n.d.). Designing a trusted client-server distributed network. [1989 Proceedings] Fifth Annual Computer Security Applications Conference. doi:10.1109/csac.1989.81033

[3] Green, P. E., Chappuis, R. J., Fisher, J. D., Frosch, P. S., & Wood, C. E. (1987). A perspective on Advanced Peer-to-Peer Networking. IBM Systems Journal, 26(4), 414–428.doi:10.1147/sj.264.0414

[4] Dongyan, Xu; Kulkarni, Sunil Suresh; Rosenberg, Catherine; and Chai, Heung-Keung. "A CDN-P2P Hybrid Architecture for Cost-Effective Streaming Media Distribution". Department of Computer Sciences, School of Electrical and Computer Engineering, Purdue University. http : // www. cs.purdue.edu/homes /dxu/pubs/MMCN03-enhanced.pdf. Published: Feb. 7, 2006.

[5] Stallings William Foundations of Modern Networking SDN, NFV, QoE, IoT, and Cloud [Book]. 800 East 96th Street, Indianapolis, Indiana 46240 USA: Pearson Education, Inc, 2015.

[6] Qusay F. Hassan (Editor) Giancarlo Fortino, Andreas Molisch, Linda Shafer Internet of Things, A TO Z, Technologies and Applications [Book]. - Hoboken, New Jersey: John Wiley & Sons, Inc., Hoboken, New Jersey, 2018.

[7] Jackson Joab, the Interplanetary Internet [Journal] // IEEE Spectrum. - August 2005.

[8] Bush Stephen F. Nanoscale Communication Networks [Book]. [s.l.] : ARTECH HOUSE, 685 Canton Street, Norwood, MA 02062, 2010.

[9] Behrouz A. Forouzan. 2003. Data Communications and Networking (3rd. ed.). McGraw-Hill, Inc., USA.

[10] Stallings William, and William Stallings. Cryptography and Network Security: Principles and Practice. Upper Saddle River, N.J.: Prentice Hall, 1999.

[11] Needham, Roger M. "Denial of service: an example." Communications of the ACM 37, no. 11 (1994): 42-46.

[12] Hancock, B. (2000). Canadian Teen Mafiaboy Pleads Guilty. Computers & Security, 19(8), 669. doi:10.1016/s0167-4048(00)08007-x

[13] Alomari, E., Manickam, S., B. Gupta, B., Karuppayah, S., & Alfaris, R. (2012). Botnet-based Distributed Denial of Service (DDoS) Attacks on Web Servers: Classification and Art. International Journal of Computer Applications, 49(7), 24–32. doi:10.5120/7640-0724

[14] Ashton. K. Internet of things. RFID J. [(2009, Jun.). Available: http: / /www.rfid journal.com/ articles/ view?4986

[15] Z. Chen, P. Yang, J. Xiong, Y. Feng and X. Li, ”TagRay: Contactless Sensing and Tracking of Mobile Objects using COTS RFID Devices,” IEEE INFOCOM 2020 - IEEE Conference on Computer Communications, Toronto, ON, Canada, 2020, pp. 307-316, doi: 10.1109/INFOCOM41043.2020.9155531.

[16] Choffnes, D. R., & Bustamante, F. E. (2005). An integrated mobility and traffic model for vehicular wireless networks. Proceedings of the 2nd ACM International Workshop on Vehicular Ad Hoc Networks - VANET ’05.doi:10.1145/1080754.1080765.

[17] Kumar G., Saha R., M. K. Rai and T. Kim, ”Multidimensional Security Provision for Secure Communication in Vehicular Ad Hoc Networks Using Hierarchical Structure and End-to-End Authentication,” in IEEE Access, vol. 6, pp. 46558-46567, 2018, doi: 10.1109/ACCESS.2018.2866759.

[18] Wang, M., Zhang, R., & Shen, X. (2015). Mobility-Aware Coordinated EV Charging in VANET-Enhanced Smart Grid. Mobile Electric Vehicles, 21–54.doi:10.1007/978-3-319-25130-1-3

[19] Saxena, N., & Choi, B. (2015).State of the Art Authentication, Access Control, and Secure Integration in Smart Grid. Energies, 8(10), 11883–11915.doi:10.3390/en81011883

[20] Li, G., Sun, Q., Boukhatem, L., Wu, J., & Yang, J. (2019).Intelligent Vehicle-to-Vehicle Charging Navigation for Mobile Electric Vehicles via VANET-Based Communication. IEEE Access, 7, 170888–170906.doi:10.1109/access.2019.2955927

[21] Vimal Bibhu,Kumar Roshan,Kumar Balwant Singh,Dhirendra Kumar Singh,”Performance Analysis of Black Hole Attack in Vanet”, IJCNIS, vol.4, no.11, pp.47-54, 2012.

[22] Hasrouny, H., Samhat, A. E., Bassil, C., & Laouiti, A. (2017).VANet security challenges and solutions: A survey. Vehicular Communications, 7, 7–20.doi:10.1016/j.vehcom.2017.01.002

[23] Biswas, S., Misic, J., & Misic, V. (2012).DDoS attack on WAVE-enabled VANET through synchronization. 2012 IEEE Global Communications Conference (GLOBECOM).doi:10.1109/glocom.2012.6503256

[24] Azer, Marianne & Gamal, Noha & Mousa, Sherif & Elkosairy, Ahmed. (2014). Jamming Attacks on VANETs.

[25] Sakiz, F., & Sen, S. (2017).A survey of attacks and detection mechanisms on intelligent transportation systems: VANETs and IoV. Ad Hoc Networks, 61, 33–50.doi:10.1016/j.adhoc.2017.03.006

[26] Mahajan, D., & Sachdeva, M. (2013). DDoS attack prevention and mitigation techniques-a review. International Journal of Computer Applications, 67(19).

[27] Feinstein, L., Schnackenberg, D., Balupari, R., & Kindred, D. (2003, April). Statistical approaches to DDoS attack detection and response. In Proceedings DARPA information survivability conference and exposition (Vol. 1, pp. 303-314). IEEE.

[28] Haider, S., Akhunzada, A., Mustafa, I., Patel, T. B., Fernandez, A., Choo, K. K. R., & Iqbal, J. (2020). A deep CNN ensemble framework for efficient DDoS attack detection in software defined networks. Ieee Access, 8, 53972-53983.

[29] B. Subba, ”A Neural Network based NIDS framework for intrusion detection in contemporary network traffic,” 2019 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS), GOA, India, 2019, pp. 1-6, doi:10.1109/ANTS47819.2019.9117966.

[30] A. Ahmim, L. Maglaras, M. A. Ferrag, M. Derdour and H. Janicke, ”A Novel Hierarchical Intrusion Detection System Based on Decision Tree and Rules-Based Models,” 2019 15th International Conference on Distributed Computing in Sensor Systems (DCOSS), Santorini Island, Greece, 2019, pp. 228-233, doi: 10.1109/DCOSS.2019.00059.

[31] Sarnovsky, M., & Paralic, J. (2020). Hierarchical Intrusion Detection Using Machine Learning and Knowledge Model. Symmetry, 12(2), 203. doi:10.3390/sym12020203

[32] N. Bui, A. P. Castellani, P. Casari and M. Zorzi, ”The internet of energy: a web-enabled smart grid system,” in IEEE Network, vol. 26, no. 4, pp. 39-45, July-August 2012, doi: 10.1109/MNET.2012.6246751.

[33] L. Gkatzikis, I. Koutsopoulos and T. Salonidis, ”The Role of Aggregators in Smart Grid Demand Response Markets,” in IEEE Journal on Selected Areas in Communications, vol. 31, no. 7, pp. 1247-1257, July 2013, doi: 10.1109/JSAC.2013.130708.

[34] Sirohi, D., Kumar, N.,& Rana, P. S. (2020). Convolutional neural networks for 5G-enabled Intelligent Transportation System: A systematic review. Computer Communications. doi:10.1016/j.comcom.2020.01.058

[35] Choffnes, D. R.,& Bustamante, F. E. (2005). An integrated mobility and traffic model for vehicular wireless networks. Proceedings of the 2nd ACM International Workshop on Vehicular Ad Hoc Networks - VANET ’05. doi:10.1145/1080754.1080765.

[36] Kumar G., R. Saha, M. K. Rai and T. Kim, ”Multidimensional Security Provision for Secure Communication in Vehicular Ad Hoc Networks Using Hierarchical Structure and End-to-End Authentication,” in IEEE Access, vol. 6, pp. 46558-46567, 2018, doi: 10.1109/ACCESS.2018.2866759.

[37] Mohammad, S. A., Rasheed, A., & Qayyum, A. (2011). VANET Architectures and Protocol Stacks: A Survey. Communication Technologies for Vehicles, 95–105. doi:10.1007/978-3-642-19786-4_9

[38] Ravi Tomar, Manish Prateek, G. H. Sastry. Vehicular Adhoc Network (VANET) - An Introduction. International Journal of Control Theory and Applications, International Science Press 2016, 9 (18), pp.8883-8888. ffhal-01496806f

[39] Balu, M., G. Kumar, and S.-J. Lim, A REVIEW ON SECURITY TECHNIQUES IN VANETS. International Journal of Control and Automation, 2019. 12(4): p. 1-14.

[40] Paranjothi, A., Khan, M. S., Nijim, M.,& Challoo, R. (2016). MAvanet: Message authentication in VANET using social networks. 2016 IEEE 7th Annual Ubiquitous Computing, Electronics& Mobile Communication Conference (UEMCON). doi:10.1109/uemcon.2016.7777915

[41] Papadimitratos, P., Buttyan, L., Holczer, T., Schoch, E., Freudiger, J., Raya, M., . . . Hubaux, J.-P. (2008). Secure vehicular communication systems: design and architecture. IEEE Communications Magazine, 46(11), 100–109. doi:10.1109/mcom.2008.4689252

[42] Engoulou, R. G., Bellaïche, M., Pierre, S.,& Quintero, A. (2014). VANET security surveys. Computer Communications, 44, 1–13. doi:10.1016/j.comcom.2014.02.020

[43] Purkait, R., Tripathi, S. Fuzzy Logic Based Multi-criteria Intelligent Forward Routing in VANET. Wireless Pers Commun 111, 1871–1897 (2020). Htt ps://doi.org/10.1007/s11277-019-06962-2

[44] Salem, F. M.,& Ali, A. S. (2020). SOS: Self-organized secure framework for VANET. International Journal of Communication Systems, e4317. doi:10.1002/dac.4317

[45] Silva, C. M., Masini, B. M., Ferrari, G.,& Thibault, I. (2017). A Survey on Infrastructure-Based Vehicular Networks. Mobile Information Systems, 2017, 1–28. doi:10.1155/2017/6123868

[46] K. Ho, P. Kang, C. Hsu, and C. Lin, “Implementation of WAVE/DSRC devices for vehicular communications,” in Proc. International Symposium on Computer, Communication, Control and Automation, vol. 2, pp. 522–525, May 2010

[47] Pauzie, A. (2002). In-vehicle communication systems: the safety aspect. Injury Prevention, 8(90004), 26iv–29. doi:10.1136/ip.8.suppl_4.iv26

[48] Neumann, A., Mytych, M. J., Wesemann, D., Wisniewski, L.,& Jasperneite, J. (2017). Approaches for In-vehicle Communication – An Analysis and Outlook. Communications in Computer and Information Science, 395–411. doi:10.1007/978-3-319-59767-6_31

[49] X. Yang, J. Liu, F. Zhao and N. Vaidya, A vehicle-to-vehicle communication protocol for cooperative collision warning, in: Proceedings of MobiQuitous’04, 2004.

[50] Ronald Jurgen, ”V2V and V2I Technical Papers,” in V2V/V2I Communications for Improved Road Safety and Efficiency , SAE, 2012, pp.19-19.

[51] J. J. Anaya, P. Merdrignac, O. Shagdar, F. Nashashibi and J. E. Naranjo, ”Vehicle to pedestrian communications for protection of vulnerable road users,” 2014 IEEE Intelligent Vehicles Symposium Proceedings, Dearborn, MI, 2014, pp. 1037-1042, doi: 10.1109/IVS.2014.6856553.

[52] Guille, C.,& Gross, G. (2009). A conceptual framework for the vehicle-to-grid (V2G) implementation. Energy Policy, 37(11), 4379–4390. doi:10.1016/j.enpol.2009.05.053

[53] Loisel, R., Pasaoglu, G.,& Thiel, C. (2014). Large-scale deployment of electric vehicles in Germany by 2030: An analysis of grid-to-vehicle and vehicle-to-grid concepts. Energy Policy, 65, 432–443. doi:10.1016/j.enpol.2013.10.029

[54] Heddebaut, M., Rioult, J., Ghys, J. P., Gransart, C.,& Ambellouis, S. (2005). Broadband vehicle-to-vehicle communication using an extended autonomous cruise control sensor. Measurement Science and Technology, 16(6), 1363–1373. doi:10.1088/0957-0233/16/6/019

[55] C. Campolo, A. Molinaro, A. Iera and F. Menichella, ”5G Network Slicing for Vehicle-to-Everything Services,” in IEEE Wireless Communications, vol. 24, no. 6, pp. 38-45, Dec. 2017, doi: 10.1109/MWC.2017.1600408.

[56] Qureshi, K. N., Abdullah, A. H., Anwar, R. W., Anwar, M.,& Awan, K. M. (2016). AEGRP: AN ENHANCED GEOGRAPHICAL ROUTING PROTOCOL FOR VANET. Jurnal Teknologi, 78(4-3). doi:10.11113/jt.v78.8257

[57] Kumar, Vishal& Chand, Narottam& Mishra, Shailendra. (2013). Applications of VANETs: Present& future. Communications and Network. 05. 10.4236/cn.2013.51B004.

[58] National Highway Traffic Safety Administration (NHTSA). Vehicle Safety COmmunications PRoject Task3Final Report: Identify Intelligent Vehicle Safety Applications Enabled by DSRC; Technical Report DOT HS; US Department of Transportation: Washington, DC, USA, 2005; pp. 809–859.

[59] Arif, M., Wang, G., Zakirul Alam Bhuiyan, M., Wang, T.,& Chen, J. (2019). A Survey on Security Attacks in VANETs: Communication, Applications and Challenges. Vehicular Communications, 100179. doi:10.1016/j.vehcom.2019.100179

[60] Mak, Tony K., Kenneth P. Laberteaux, and Raja Sengupta. ”A multi-channel VANET providing concurrent safety and commercial services.” Proceedings of the 2nd ACM international workshop on Vehicular ad hoc networks. 2005.

[61] Hartenstein, Hannes, and Kenneth Laberteaux, eds. VANET: vehicular applications and inter-networking technologies. Vol. 1. John Wiley& Sons, 2009.

[62] Sadatpour, V., Zargari, F.& Ghanbari, M. A Collision Aware Opportunistic Routing Protocol for VANETs in Highways. Wireless Pers Commun 109, 175–188 (2019). Ht tps://doi.org/10.1007/s11277-019-06557-x

[63] Zhu, Liehuang, et al. ”Traffic Monitoring in Self-Organizing VANETs: A Privacy-Preserving Mechanism for Speed Collection and Analysis.” IEEE Wireless Communications 26.6 (2019): 18-23.

[64] Englund, C., Chen, L., Vinel, A.,& Lin, S. Y. (2015). Future Applications of VANETs. Vehicular Ad Hoc Networks, 525–544. doi:10.1007/978-3-319-15497-8_ 18.

[65] Kumar, G., Rai, M., Saha, R., Buchanan, W. J., Thomas, R., Geetha, G.,& Rodrigues, J. (2020). A Privacy-Preserving Secure Framework for Electric Vehicles in IoT using Matching Market and Signcryption. IEEE Transactions on Vehicular Technology.

[66] Wang, Miao& Liang, Hao& Deng, Ruilong& Zhang, Ran& Shen, Xuemin. (2013). VANET based online charging strategy for electric vehicles. 4804-4809. 10.1109/GLOCOMW.2013.6855711.

[67] R. G. Gago, S. F. Pinto and J. F. Silva, ”G2V and V2G electric vehicle charger for smart grids,” 2016 IEEE International Smart Cities Conference (ISC2), Trento, 2016, pp. 1-6, doi: 10.1109/ISC2.2016.7580786.

[68] Liu, J., Xiao, Y., Li, S., Liang, W.,& Chen, C. P. (2012). Cyber security and privacy issues in smart grids. IEEE Communications Surveys& Tutorials, 14(4), 981-997.

[69] Stallings, William, and William Stallings. Cryptography and Network Security: Principles and Practice. Upper Saddle River, N.J.: Prentice Hall, 1999.

[70] Mokthar, B.,& Azab, M. (2015). Survey on security issues in vehicular ad hoc networks. Alexandria Engineering Journal, 54, 1115–1126

[71] Pathre, A., Agrawal, C.,& Jain, A. (2013). Identification of malicious vehicle in vanet environment from ddos attack. International Journal of Global Research in Computer Science, 4(6), 30–34.

[72] Gamal, Manal S., Abdurrahman A. Nasr, and Sayed A. Nouh. ”VANET SECURITY: DEFENSE AND DETECTION, A REVIEW.” Journal of Al-Azhar University Engineering Sector 15.56 (2020): 810-827.

[73] Ahmad, F., Adnane, A., Franqueira, V. N., Kurugollu, F.,& Liu, L. (2018). Man-in-the-middle attacks in vehicular ad-hoc networks: evaluating the impact of attackers' strategies. Sensors, 18(11), 4040.

[74] Li, W., Wen, Q., Su, Q.,& Jin, Z. (2012). An efficient and secure mobile payment protocol for restricted connectivity scenarios in vehicular ad hoc network. Computer Communications, 35(2), 188–195. doi:10.1016/j.comcom.2011.09.003

[75] Cencioni P., Di Pietro R.: 'A mechanism to enforce privacy in vehicle-to-infrastructure communication', Comput. Commun., 2008, 31, (12), pp. 2790– 2802

[76] Sumra, I. A., Ahmad, I.,& Hasbullah, H. (2011, April). Classes of attacks in VANET. In 2011 Saudi International Electronics, Communications and Photonics Conference (SIECPC) (pp. 1-5). IEEE.

[77] D. P. Choudhari and S. S. Dorle, ”Maximization of packet delivery ratio for DADCQ protocol after removal of Eavesdropping and DDoS attacks in VANET,” 2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Kanpur, India, 2019, pp. 1-8, doi: 10.1109/ICCCNT45670.2019.8944485.

[78] Malhi, A. K.,& Batra, S. (2016). Genetic-based framework for prevention of masquerade and DDoS attacks in vehicularad-hocnetworks. Security and Communication Networks, 9(15), 2612–2626. doi:10.1002/sec.1506

[79] Hezam Al Junaid, M. A., Syed, A. A., Mohd Warip, M. N., Fazira Ku Azir, K. N.,& Romli, N. H. (2018). Classification of Security Attacks in VANET: A Review of Requirements and Perspectives. MATEC Web of Conferences, 150, 06038. doi:10.1051/matecconf/201815006038

[80] Malik, K. M., Malik, H.,& Baumann, R. (2019). Towards Vulnerability Analysis of Voice-Driven Interfaces and Countermeasures for Replay Attacks. 2019 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR). doi:10.1109/mipr.2019.00106

[81] Singh, K.,& Sharma, S. Advanced Security Attacks on Vehicular AD HOC Network (VANET). International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, Volume-9 Issue-2, December 2019

[82] N. Lo and H. Tsai, ”Illusion Attack on VANET Applications - A Message Plausibility Problem,” 2007 IEEE Globecom Workshops, Washington, DC, 2007, pp. 1-8, doi: 10.1109/GLOCOMW.2007.4437823.

[83] Komal B. Sahare, DR. L G. Malik, “An approach for detection of attack in VANET”, International journal of engineering research ‘and application (IJERA) and International conference on industrial automation and computing (ICIAC), 2014.

[84] Alomari, E., Manickam, S., B. Gupta, B., Karuppayah, S.,& Alfaris, R. (2012). Botnet-based Distributed Denial of Service (DDoS) Attacks on Web Servers: Classification and Art. International Journal of Computer Applications, 49(7), 24–32. doi:10.5120/7640-0724

[85] Vikash Porwal, Rajeev Patel and Dr. R.K. Kapoor, “An investigation of DoS flooding attack in VANET”, International journal of advance foundation and research in computing (IJAFRC), Vol. 1, Dec 2014.

[86] Vimal Bibhu,Kumar Roshan,Kumar Balwant Singh,Dhirendra Kumar Singh,”Performance Analysis of Black Hole Attack in Vanet”, IJCNIS, vol.4, no.11, pp.47-54, 2012.

[87] Hasrouny, H., Samhat, A. E., Bassil, C.,& Laouiti, A. (2017). VANet security challenges and solutions: A survey. Vehicular Communications, 7, 7–20. doi:10.1016/j.vehcom.2017.01.002

[88] Azer, Marianne& Gamal, Noha& Mousa, Sherif& Elkosairy, Ahmed. (2014). Jamming Attacks on VANETs.

[89] Sakiz, F.,& Sen, S. (2017). A survey of attacks and detection mechanisms on intelligent transportation systems: VANETs and IoV. Ad Hoc Networks, 61, 33–50. doi:10.1016/j.adhoc.2017.03.006

[90] D. Kshirsagar and A. Patil, ”Blackhole attack detection and prevention by real time monitoring,” 2013 Fourth International Conference on Computing, Communications and Networking Technologies (ICCCNT), Tiruchengode, 2013, pp. 1-5, doi: 10.1109/ICCCNT.2013.6726597.

[91] Sen, J., Chandra, M. G., Harihara, S. G., Reddy, H.,& Balamuralidhar, P. (2007, December). A mechanism for detection of gray hole attack in mobile Ad Hoc networks. In 2007 6th International Conference on Information, Communications& Signal Processing (pp. 1-5). IEEE.

[92] Mejri, M. N.,& Ben-Othman, J. (2014). Entropy as a new metric for denial of service attack detection in vehicular ad-hoc networks. Proceedings of the 17th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems - MSWiM '14. doi:10.1145/2641798.2641800

[93] R. John, J. P. Cherian and J. J. Kizhakkethottam, "A survey of techniques to prevent sybil attacks," 2015 International Conference on Soft-Computing and Networks Security (ICSNS), Coimbatore, 2015, pp. 1-6, doi: 10.1109/ICSNS.2015.7292385.

[94] Douceur J.R. (2002) The Sybil Attack. In: Druschel P., Kaashoek F., Rowstron A. (eds) Peer-to-Peer Systems. IPTPS 2002. Lecture Notes in Computer Science, vol 2429. Springer, Berlin, Heidelberg. htt ps://doi.org/10.1007/3-540-45748-8_ 24

[95] Sheikh, M. S., Liang, J.,& Wang, W. (2019). A survey of security services, attacks, and applications for vehicular ad hoc networks (vanets). Sensors, 19(16), 3589.

[96] Shilpa, P.,& Patil, R. B. (2015). Cooperative message authentication and resisting free riding attacks in VANETS. Int J Res Eng Technol, 4(5), 127-131.

[97] Li, J., Lu, H.,& Guizani, M. (2014). ACPN: A novel authentication framework with conditional privacy-preservation and non-repudiation for VANETs. IEEE Transactions on Parallel and Distributed Systems, 26(4), 938-948.

[98] Leinmuller, T., Schmidt, R. K., Schoch, E., Held, A.,& Schafer, G. (2008). Modeling Roadside Attacker Behavior in VANETs. 2008 IEEE Globecom Workshops. doi:10.1109/glocomw.2008.ecp.63

[99] Upadhyaya, A. N.,& Shah, J. S. (2018). Attacks on vanet security. Int J Comp Eng Tech, 9(1), 8-19.

[100] Liang W, Li Z, Zhang H, Wang S, Bie R. Vehicular Ad Hoc Networks: Architectures, Research Issues, Methodologies, Challenges, and Trends. International Journal of Distributed Sensor Networks. August 2015. doi:10.1155/2015/745303

[101] Wang, Y., Ding, Z., Li, F., Xia, X.,& Li, Z. (2017). Design and implementation of a VANET application complying with WAVE protocol. 2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET). doi:10.1109/wispnet.2017.8300177

[102] Shah, J. C., Patel, K., Trapasiya, S. D.,& Rathod, J. M. (2014). Study for Implementation of VANET with Transport Layer Protocol. International Journal of Computer Applications, 975, 8887.

[103] Ilavendhan, A.,& Saruladha, K. (2018). Comparative study of game theoretic approaches to mitigate network layer attacks in VANETs. Ict Express, 4(1), 46-50.

[104] Khan, U. A.,& Lee, S. S. Multi-Layer Problems and Solutions in VANETs: A. Electronics 2019, 8, 204; doi:10.3390/electronics8020204

[105] P. Tyagi and D. Dembla, ”Investigating the security threats in Vehicular ad hoc Networks (VANETs): Towards security engineering for safer on-road transportation,” 2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI), New Delhi, 2014, pp. 2084-2090, doi: 10.1109/ICACCI.2014.6968313.

[106] Mansour, M. B., Salama, C., Mohamed, H. K.,& Hammad, S. A. (2018). VANET Security and Privacy-An Overview. International Journal of Network Security& Its Applications (IJNSA) Vol, 10.

[107] J. Kong, X. Hong and M. Gerla, “A new set of passive routing attack in Mobile ad hoc networks”, Proc. IEEE Military Communication Conference MILCOM, October 2003

[108] D. Karagiannis and A. Argyriou, “Jamming attack detection in a pair of RF communicating vehicles using unsupervised machine learning,” Veh. Commun., vol. 13, pp. 56–63, 2018.

[109] Tolba, Amr. (2018). Trust-Based Distributed Authentication Method for Collision Attack Avoidance in VANETs. IEEE Access. PP. 1-1. 10.1109/ACCESS.2018.2875906.

[110] Bhatt, Mayank& Sharma, Shabnam& Prakash, Aditya& Pandey, U.S.& Jyoti, Kiran. (2016). Traffic Collision Avoidance in VANET Using Computational Intelligence. International Journal of Engineering and Technology. 8. 364-370.

[111] Rawat, A., Sharma, S.,& Sushil, R. (2012). VANET: Security attacks and its possible solutions. Journal of Information and Operations Management, 3(1), 301.

[112] Oka, Dennis& Larson, Ulf. (2009). A Defense-in-Depth Approach to Securing the Wireless Vehicle Infrastructure. Journal of Networks. 4. 10.4304/jnw.4.7.552-564.

[113] W. Whyte, A. Weimerskirch, V. Kumar, T. Hehn, “A Security Credential Management System for V2V Communications”, IEEE Vehicular Networking Conference, 2013.

[114] Wang, M., Zhang, R.,& Shen, X. (2015). Mobility-Aware Coordinated EV Charging in VANET-Enhanced Smart Grid. Mobile Electric Vehicles, 21–54. doi:10.1007/978-3-319-25130-1_3

[115] Saxena, N.,& Choi, B. (2015). State of the Art Authentication, Access Control, and Secure Integration in Smart Grid. Energies, 8(10), 11883–11915. doi:10.3390/en81011883

[116] Li, G., Sun, Q., Boukhatem, L., Wu, J.,& Yang, J. (2019). Intelligent Vehicle-to-Vehicle Charging Navigation for Mobile Electric Vehicles via VANET-Based Communication. IEEE Access, 7, 170888–170906. doi:10.1109/access.2019.2955927

[117] Falk, R.,& Fries, S. (2012). Electric vehicle charging infrastructure security considerations and approaches. Proc. of INTERNET, 58-64.

[118] Tanwar, S, Vora, J, Tyagi, S, Kumar, N, Obaidat, MS. A systematic review on security issues in vehicular ad hoc network. Security and Privacy 2018; 1:e39. ht tps://doi.org/10.1002/spy2.39

[119] D. Boneh and M. Franklin, "Identity-Based Encryption from the Weil Pairings," Advances in Cryptology-Asiacrypt, Springer-Verlag, pp. 514-532, 2001.

[120] Zhang, F.,& Kim, K. (2002, December). ID-based blind signature and ring signature from pairings. In International Conference on the Theory and Application of Cryptology and Information Security (pp. 533-547). Springer, Berlin, Heidelberg.

[121] Choon, J. C.,& Hee Cheon, J. (2002). An Identity-Based Signature from Gap Diffie-Hellman Groups. Lecture Notes in Computer Science, 18–30. doi:10.1007/3-540-36288-6_ 2

[122] Chow, S. S. M., Yiu, S.-M.,& Hui, L. C. K. (2005). Efficient Identity Based Ring Signature. Lecture Notes in Computer Science, 499–512. doi:10.1007/11496137_34

[123] Gamage, C., Gras, B., Crispo, B.,& Tanenbaum, A. S. (2006, August). An identity-based ring signature scheme with enhanced privacy. In 2006 Securecomm and Workshops (pp. 1-5). IEEE.

[124] Kamat, P., Baliga, A.,& Trappe, W. (2006). An identity-based security framework For VANETs. Proceedings of the 3rd International Workshop on Vehicular Ad Hoc Networks - VANET '06. doi:10.1145/1161064.1161083

[125] J. Sun, C. Zhang, Y. Zhang and Y. Fang, "An Identity-Based Security System for User Privacy in Vehicular Ad Hoc Networks," in IEEE Transactions on Parallel and Distributed Systems, vol. 21, no. 9, pp. 1227-1239, Sept. 2010, doi: 10.1109/TPDS.2010.14.

[126] Lim, H. W.,& Paterson, K. G. (2010). Identity-based cryptography for grid security. International Journal of Information Security, 10(1), 15–32. doi:10.1007/s10207-010-0116-z

[127] D. He, S. Zeadally, B. Xu and X. Huang, "An Efficient Identity-Based Conditional Privacy-Preserving Authentication Scheme for Vehicular Ad Hoc Networks," in IEEE Transactions on Information Forensics and Security, vol. 10, no. 12, pp. 2681-2691, Dec. 2015, doi: 10.1109/TIFS.2015.2473820.

[128] Ali, I., Lawrence, T.,& Li, F. (2019). An Efficient Identity-Based Signature Scheme without Bilinear Pairing for Vehicle-To-Vehicle Communication in VANETs. Journal of Systems Architecture, 101692. doi:10.1016/j.sysarc.2019.101692

[129] Limbasiya, T.,& Das, D. (2019). Identity based proficient message verification scheme for vehicle users. Pervasive and Mobile Computing, 60, 101083. doi:10.1016/j.pmcj.2019.101083

[130] M. A. Al-Shareeda, M. Anbar, S. Manickam and A. A. Yassin, "VPPCS: VANET-Based Privacy-Preserving Communication Scheme," in IEEE Access, vol. 8, pp. 150914-150928, 2020, doi: 10.1109/ACCESS.2020.3017018.

[131] Paterson, K. G.,& Price, G. (2003). A comparison between traditional public key infrastructures and identity-based cryptography. Information Security Technical Report, 8(3), 57–72. doi:10.1016/s1363-4127(03)00308-x

[132] K. Sanzgiri, B. Dahill, B. N. Levine, C. Shields and E. M. Belding-Royer, ”A secure routing protocol for ad hoc networks,” 10th IEEE International Conference on Network Protocols, 2002. Proceedings., Paris, France, 2002, pp. 78-87, doi: 10.1109/ICNP.2002.1181388.

[133] Y. C. Hu,A. Perrig, D. B. Johnson, ”Ariadne: A Secure On-Demand Routing Protocol for Ad Hoc Networks”, MobiCom’02, pp. 23-26, 2002

[134] Hu, Y.-C., Johnson, D. B., & Perrig, A. (2003). SEAD: secure efficient distance vector routing for mobile wireless ad hoc networks. Ad Hoc Networks, 1(1), 175–192. doi:10.1016/s1570-8705(03)00019-2

[135] Poongodi, M., Hamdi, M., Sharma, A., Ma, M., & Singh, P. K. (2019). DDoS Detection Mechanism Using Trust-Based Evaluation System in VANET. IEEE Access, 7, 183532–183544. doi:10.1109/access.2019.2960367

[136] T. Nandy, R. M. Noor, M. Yamani Idna Bin Idris and S. Bhattacharyya, ”T-BCIDS: Trust-Based Collaborative Intrusion Detection System for VANET,” 2020 National Conference on Emerging Trends on Sustainable Technology and Engineering Applications (NCETSTEA), Durgapur, India, 2020, pp. 1-5, doi: 10.1109/NCETSTEA48365.2020.9119934.

[137] J. Grover, N. K. Prajapati, V. Laxmi, and M. S. Gaur, “Machine learning approach for multiple misbehavior detection in VANET,” Commun. Comput. Inf. Sci., vol. 192 CCIS, no. PART 3, pp. 644–653, 2011.

[138] W. Li, A. Joshi, and T. Finin, “SVM-CASE: An SVM-based context aware security framework for vehicular ad-hoc networks,” 2015 IEEE 82nd Veh. Technol. Conf. VTC Fall 2015 -Proc., pp. 1–5, 2015.

[139] F. A. Ghaleb and F. Mohammed, “An Effective Misbehavior Detection Model using Artificial Neural Network for Vehicular Ad hoc Network Applications,” pp. 13–18, 2017.

[140] M. Kim, I. Jang, S. Choo, J. Koo, and S. Pack, “Collaborative security attack detection in software-defined vehicular networks,” 19th Asia-Pacific Netw. Oper. Manag. Symp. Manag. a World Things, APNOMS 2017, pp. 19–24, 2017.

[141] Y. A. O. Yu, L. E. I. Guo, Y. E. Liu, J. Zheng, and Y. U. E. Zong, “An Efficient SDN-Based DDoS Attack Detection and Rapid Response Platform in Vehicular Networks,” IEEE Access, vol. 6, pp. 44570–44579, 2018.

[142] Liang, J., Chen, J., Zhu, Y., & Yu, R. (2018). A novel Intrusion Detection System for Vehicular Ad Hoc Networks (VANETs) based on differences of traffic flow and position. Applied Soft Computing. doi:10.1016/j.asoc.2018.12.001

[143] Kosmanos, D., Pappas, A., Maglaras, L., Moschoyiannis, S., Aparicio-Navarro, F. J., Argyriou, A., & Janicke, H. (2019). A novel Intrusion Detection System against spoofing attacks in connected Electric Vehicles. Array, 100013. doi:10.1016/j.array.2019.100013

[144] Kaur, J., Singh, T., & Lakhwani, K. (2019). An Enhanced Approach for Attack Detection in VANETs Using Adaptive Neuro-Fuzzy System. 2019 International Conference on Automation, Computational and Technology Management (ICACTM). doi:10.1109/icactm.2019.8776833

[145] Aloqaily, M., Otoum, S., Ridhawi, I. A., & Jararweh, Y. (2019). An Intrusion Detection System for Connected Vehicles in Smart Cities. Ad Hoc Networks. doi:10.1016/j.adhoc.2019.02.001

[146] Kolandaisamy, R., Noor, R. M., Zaba, M. R., Ahmedy, I., & Kolandaisamy, I. (2019). Markov Chain Based Ant Colony Approach for Mitigating DDoS Attacks Using Integrated Vehicle Mode Analysis in VANET. 2019 IEEE 1st International Conference on Energy, Systems and Information Processing (ICESIP). doi:10.1109/icesip46348.2019.8938253

[147] Zeng, Y., Qiu, M., Zhu, D., Xue, Z., Xiong, J., & Liu, M. (2019). DeepVCM: A Deep Learning Based Intrusion Detection Method in VANET. 2019 IEEE 5th Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing, (HPSC) and IEEE Intl Conference on Intelligent Data and Security (IDS). doi:10.1109/bigdatasecurity-hpsc-ids.2019.00060

[148] P. Manimaran and A. R. K. P, ”NDNIDS: An Intrusion Detection System for NDN Based VANET,” 2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring), Antwerp, Belgium, 2020, pp. 1-5, doi: 10.1109/VTC2020-Spring48590.2020.9129365.

[149] Shahverdy, M., Fathy, M., Berangi, R., & Sabokrou, M. (2020). Driver Behavior Detection and Classification Using Deep Convolutional Neural Networks. Expert Systems with Applications, 113240. doi:10.1016/j.eswa.2020.113240

[150] Schmidt, D. A., Khan, M. S., & Bennett, B. T. (2020). Spline-based intrusion detection for VANET utilizing knot flow classification. Internet Technology Letters. doi:10.1002/itl2.155

[151] Adhikary, K., Bhushan, S., Kumar, S., & Dutta, K. (2020). Hybrid Algorithm to Detect DDoS Attacks in VANETs. Wireless Personal Communications. doi:10.1007/s11277-020-07549-y .

[152] Liu, T., Shi, S., & Gu, X. (2020). Naive Bayes Classifier Based Driving Habit Prediction Scheme for VANET Stable Clustering. Mobile Networks and Applications. doi:10.1007/s11036-020-01580-w

[153] Lahrouni, Y., Pereira, C., Bensaber, B. A., & Biskri, I. (2017). Using Mathematical Methods Against Denial of Service (DoS) Attacks in VANET. Proceedings of the 15th ACM International Symposium on Mobility Management and Wireless Access - MobiWac ’17. doi:10.1145/3132062.3132065

[154] G. Li, X. Li, Q. Sun, L. Boukhatem and J. Wu, ”An Effective MEC Sustained Charging Data Transmission Algorithm in VANET-Based Smart Grids,” in IEEE Access, vol. 8, pp. 101946-101962, 2020, doi:10.1109/ACCESS.2020.2998018.

[155] Z. Wan, W.-T. Zhu, and G. Wang, “Prac: Efficient privacy protection for vehicle-to-grid communications in the smart grid,” Computers & security, vol. 62, pp. 246–256, 2016

[156] H. Liu, Y. Zhang and T. Yang, ”Blockchain-Enabled Security in Electric Vehicles Cloud and Edge Computing,” in IEEE Network, vol. 32, no. 3, pp. 78-83, May/June 2018, doi: 10.1109/MNET.2018.1700344.

[157] M. Kim, K. Park, S. Yu, J. Lee, Y. Park, S.-W. Lee, and B. Chung, “A secure charging system for electric vehicles based on blockchain,” Sensors, vol. 19, no. 13, p. 3028, 2019.

[158] H. Marzougui, A. Kadri, J.-P. Martin, M. Amari, S. Pierfederici, and F. Bacha, “Implementation of energy management strategy of hybrid power source for electrical vehicle,” Energy Conversion and Management, vol. 195, pp. 830–843, 2019.

[159] A. Kavousi-Fard, T. Jin, W. Su and N. Parsa, ”An Effective Anomaly Detection Model for Securing Communications in Electric Vehicles,” in IEEE Transactions on Industry Applications, doi: 10.1109/TIA.2020.3005062.

[160] W. Zhijun, W. Jingjie and Y. Meng, ”Prevention of DoS Attacks in Information-Centric Networking,” 2018 IEEE Conference on Application, Information and Network Security (AINS), Langkawi, Malaysia, 2018, pp. 105-110, doi: 10.1109/AINS.2018.8631473.

[161] L. Feinstein, D. Schnackenberg, R. Balupari and D. Kindred, ”Statistical approaches to DDoS attack detection and response,” Proceedings DARPA Information Survivability Conference and Exposition, Washington, DC, USA, 2003, pp. 303-314 vol.1, doi: 10.1109/DISCEX.2003.1194894.

[162] R. Sharma, C. A. Chan and C. Leckie, ”Evaluation of Centralised vs Distributed Collaborative Intrusion Detection Systems in Multi-Access Edge Computing,” 2020 IFIP Networking Conference (Networking), Paris, France, 2020, pp. 343-351.

[163] M. Kumar and A. K. Singh, ”Distributed Intrusion Detection System using Blockchain and Cloud Computing Infrastructure,” 2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184), Tirunelveli, India, 2020, pp. 248-252, doi: 10.1109/ICOEI48184.2020.9142954.

[164] Hu, J., Yu, X., Qiu, D., & Chen, H.-H. (2009). A simple and efficient hidden Markov model scheme for host-based anomaly intrusion detection. IEEE Network, 23(1), 42–47.

[165] Introduction to Intrusion Detection Systems. (2003). Cisco Security Professional’s Guide to Secure Intrusion Detection Systems, 1–38.

[166] R. Venkatesan, D. R. Devi, R. Keerthana and A. A. Kumar, ”A NOVEL APPROACH FOR DETECTING DDoS ATTACK IN H-IDS USING ASSOCIATION RULE,” 2018 IEEE International Conference on System, Computation, Automation and Networking (ICSCA), Pondicherry, 2018, pp. 1-5, doi: 10.1109/ICSCAN.2018.8541174.

[167] Besharati, E., Naderan, M. & Namjoo, E. LR-HIDS: logistic regression host-based intrusion detection system for cloud environments. J Ambient Intell Human Comput 10, 3669–3692 (2019).

[168] K. Rahul Vigneswaran, Prabaharan Poornachandran, and KP Soman, “A Compendium on Network and Host Based Intrusion Detection Systems.” Kumar, A., Paprzycki, M., & Gunjan, V. K. (Eds.). (2020). ICDSMLA 2019. Lecture Notes in Electrical Engineering. doi:10.1007/978-981-15-1420-3

[169] Saha, Rahul, Gulshan Kumar, Mritunjay Kumar Rai, Reji Thomas, and Se-Jung Lim. ”Privacy Ensured e-healthcare for fog-enhanced IoT based applications.” IEEE Access 7 (2019): 44536-44543.

[170] Chen, Ziyang, Panlong Yang, Jie Xiong, Yuanhao Feng, and Xiang-Yang Li. ”Tagray: Contactless sensing and tracking of mobile objects using cots RFID devices.” In IEEE INFOCOM 2020-IEEE Conference on Computer Communications, pp. 307-316. IEEE, 2020.

[171] Verma, A., Saha, R., Kumar, G., & Kim, T. H. (2021). The security perspectives of vehicular networks: a taxonomical analysis of attacks and solutions. Applied Sciences, 11(10), 4682.

[172] Kumar, Gulshan, Rahul Saha, Mritunjay Kumar Rai, and Tai-Hoon Kim. ”Multidimensional security provision for secure communication in vehicular ad hoc networks using hierarchical structure and end-to-end authentication.” IEEE Access 6 (2018): 46558-46567.

[173] WKumar, G., Saha, R., Rai, M. K., Buchanan, W. J., Thomas, R., Geetha, G., ... & Rodrigues, J. J. (2020). A privacy-preserving secure framework for electric vehicles in IoT using matching market and signcryption. IEEE Transactions on Vehicular Technology, 69(7), 7707-7722.

[174] Kumar, Ashish, Rahul Saha, Mamoun Alazab, and Gulshan Kumar. ”A lightweight signcryption method for perception layer in Internet-of-Things.” Journal of Information Security and Applications 55 (2020): 102662.

[175] Li, Guangyu, Qiang Sun, Lila Boukhatem, Jinsong Wu, and Jian Yang. ”Intelligent vehicle-to-vehicle charging navigation for mobile electric vehicles via VANET-based communication.” IEEE Access 7 (2019): 170888-170906.

[176] Inedjaren, Youssef, Mohamed Maachaoui, Besma Zeddini, and Jean-Pierre Barbot. ”Blockchain-based distributed management system for trust in VANET.” Vehicular Communications 30 (2021): 100350.

[177] Khalaf, Bashar Ahmad, Salama A. Mostafa, Aida Mustapha, Mazin Abed Mohammed, Moamin A. Mahmoud, Bander Ali Saleh Al-Rimy, Shukor Abd Razak, Mohamed Elhoseny, and Adam Marks. ”An adaptive protection of flooding attacks model for complex network environments.” Security and Communication Networks 2021 (2021).

[178] Kshirsagar, Deepak, and Sandeep Kumar. ”A feature reduction based reflected and exploited DDoS attacks detection system.” Journal of Ambient Intelligence and Humanized Computing 13, no. 1 (2022): 393-405.

[179] Naqvi, Ila, Alka Chaudhary, and Ajay Rana. ”Intrusion Detection in VANETs.” In 2021 9th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO), pp. 1-5. IEEE, 2021.

[180] Hasrouny, H., A. E. Samhat, C. Bassil, and A. Laouiti. ”VANET security challenges and solutions: a survey. Veh. Commun. 7, 7–20 (2017).”

[181] Othman, Wajdy, Miao Fuyou, Kaiping Xue, and Ammar Hawbani. ”Physically secure lightweight and privacy-preserving message authentication protocol for VANET in smart city.” IEEE Transactions on Vehicular Technology 70, no. 12 (2021): 12902-12917.

[182] Reddy, Ganesh. ”A Delay Sensitive Multi-Path Selection to Prevent the Rushing Attack in VANET.” In 2021 5th International Conference on Information Systems and Computer Networks (ISCON), pp. 1-7. IEEE, 2021.

[183] Sakiz, Fatih, and Sevil Sen. ”A survey of attacks and detection mechanisms on intelligent transportation systems: VANETs and IoV.” Ad Hoc Networks 61 (2017): 33-50.

[184] Grover, Jyoti, Nitesh Kumar Prajapati, Vijay Laxmi, and Manoj Singh Gaur. ”Machine learning approach for multiple misbehavior detection in VANET.” In International conference on advances in computing and communications, pp. 644-653. Springer, Berlin, Heidelberg, 2011.

[185] Li, Wenjia, Anupam Joshi, and Tim Finin. ”SVM-CASE: An SVM-based context aware security framework for vehicular ad-hoc networks.” In 2015 IEEE 82nd Vehicular Technology Conference (VTC2015-Fall), pp. 1-5. IEEE, 2015.

[186] Ghaleb, Fuad A., Anazida Zainal, Murad A. Rassam, and Fathey Mohammed. ”An effective misbehavior detection model using artificial neural network for vehicular ad hoc network applications.” In 2017 IEEE conference on application, information and network security (AINS), pp. 13-18. IEEE, 2017.

[187] Kim, Myeongsu, Insun Jang, Sukjin Choo, Jungwoo Koo, and Sangheon Pack. ”Collaborative security attack detection in software-defined vehicular networks.” In 2017 19th Asia-Pacific network operations and management symposium (APNOMS), pp. 19-24. IEEE, 2017.

[188] Yu, Yao, Lei Guo, Ye Liu, Jian Zheng, and Y. U. E. Zong. ”An efficient SDN-based DDoS attack detection and rapid response platform in vehicular networks.” IEEE Access 6 (2018): 44570-44579.

[189] Karagiannis, Dimitrios, and Antonios Argyriou. ”Jamming attack detection in a pair of RF communicating vehicles using unsupervised machine learning.” vehicular communications 13 (2018): 56-63.

[190] Liang, Junwei, Jianyong Chen, Yingying Zhu, and Richard Yu. ”A novel Intrusion Detection System for Vehicular Ad Hoc Networks (VANETs) based on differences of traffic flow and position.” Applied Soft Computing 75 (2019): 712-727.

[191] Kosmanos, Dimitrios, Apostolos Pappas, Leandros Maglaras, Sotiris Moschoyiannis, Francisco J. Aparicio-Navarro, Antonios Argyriou, and Helge Janicke. ”A novel intrusion detection system against spoofing attacks in connected electric vehicles.” Array 5 (2020): 100013.

[192] Kaur, Jasleen, Tejpreet Singh, and Kamlesh Lakhwani. ”An enhanced approach for attack detection in vanets using adaptive neuro-fuzzy system.” In 2019 International Conference on Automation, Computational and Technology Management (ICACTM), pp. 191-197. IEEE, 2019.

[193] Aloqaily, Moayad, Safa Otoum, Ismaeel Al Ridhawi, and Yaser Jararweh. ”An intrusion detection system for connected vehicles in smart cities.” Ad Hoc Networks 90 (2019): 101842.

[194] Kolandaisamy, Raenu, Rafidah Md Noor, Muhammad Reza Zaba, Ismail Ahmedy, and Indraah Kolandaisamy. ”Markov chain based ant colony approach for mitigating DDoS attacks using integrated vehicle mode analysis in VANET.” In 2019 IEEE 1st International Conference on Energy, Systems and Information Processing (ICESIP), pp. 1-5. IEEE, 2019.

[195] Manimaran, Praveensankar. ”NDNIDS: An intrusion detection system for NDN based VANET.” In 2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring), pp. 1-5. IEEE, 2020.

[196] Schmidt, David A., Mohammad S. Khan, and Brian T. Bennett. ”Spline-based intrusion detection for VANET utilizing knot flow classification.” Internet Technology Letters 3, no. 3 (2020): e155.

[197] Adhikary, Kaushik, Shashi Bhushan, Sunil Kumar, and Kamlesh Dutta. ”Hybrid algorithm to detect DDoS attacks in VANETs.” Wireless Personal Communications 114, no. 4 (2020): 3613-3634.

[198] Kadam, Nivedita, and Raja Sekhar Krovi. ”Machine Learning Approach of Hybrid KSVN Algorithm to Detect DDoS Attack in VANET.” International Journal of Advanced Computer Science and Applications 12, no. 7 (2021).

[199] Türkoğlu, Muammer, Hüseyin Polat, Cemal Koçak, and Onur Polat. ”Recognition of DDoS Attacks on SD-VANET Based on Combination of Hyperparameter Optimization and Feature Selection.” Expert Systems with Applications (2022): 117500.

[200] Shen, Wong Yan, Selvakumar Manickam, and Mahmood A. Al-Shareeda. ”Review of Advanced Monitoring Mechanisms in Peer-to-Peer (P2P) Botnets.” arXiv preprint arXiv:2207.12936 (2022).

[201] Yusof, A. R. A., Udzir, N. I., Selamat, A., Hamdan, H., & Abdullah, M. T. (2017, November). Adaptive feature selection for denial of services (DoS) attack. In 2017 IEEE Conference on Application, Information and Network Security (AINS) (pp. 81-84). IEEE.

[202] Amrita, Megha Aggarwal. ”Performance analysis of different feature selection methods in intrusion detection.” (2013).

[203] Zang, Mingyuan, and Ying Yan. ”Machine learning-based intrusion detection system for big data analytics in VANET.” In 2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring), pp. 1-5. IEEE, 2021.

[204] Bangui, Hind, Mouzhi Ge, and Barbora Buhnova. ”A hybrid data-driven model for intrusion detection in VANET.” Procedia Computer Science 184 (2021): 516-523.

[205] Gonçalves, Fábio, Joaquim Macedo, and Alexandre Santos. ”An Intelligent Hierarchical Security Framework for VANETs.” Information 12, no. 11 (2021): 455.

[206] https://zenodo.org/record/5567417

[207] Soni, Gaurav, Kamlesh Chandravanshi, Mahendra Ku Jhariya, and Arjun Rajput. ”An IPS approach to secure V-RSU communication from blackhole and wormhole attacks in VANET.” In Contemporary Issues in Communication, Cloud and Big Data Analytics, pp. 57-65. Springer, Singapore, 2022.

[208] Malik, Abdul, Muhammad Zahid Khan, Mohammad Faisal, Faheem Khan, and Jung-Taek Seo. ”An Efficient Dynamic Solution for the Detection and Prevention of Black Hole Attack in VANETs.” Sensors 22, no. 5 (2022): 1897.

[209] Mahajan, Deepika, and Monika Sachdeva. ”DDoS attack prevention and mitigation techniques-a review.” International Journal of Computer Applications 67, no. 19 (2013).

[210] Ferguson, Paul, and Daniel Senie. ”rfc2827: network ingress filtering: defeating denial of service attacks which employ ip source address spoofing.” (2000).

[211] Park, Kihong, and Heejo Lee. ”On the effectiveness of route-based packet filtering for distributed DoS attack prevention in power-law internets.” ACM SIGCOMM computer communication review 31, no. 4 (2001): 15-26.

[212] Peng, Tao, Christopher Leckie, and Kotagiri Ramamohanarao. ”Protection from distributed denial of service attacks using history-based IP filtering.” In IEEE International Conference on Communications, 2003. ICC’03., vol. 1, pp. 482-486. IEEE, 2003..

[213] Jin, Cheng, Haining Wang, and Kang G. Shin. ”Hop-count filtering: an effective defense against spoofed DDoS traffic.” In Proceedings of the 10th ACM conference on Computer and communications security, pp. 30-41. 2003.

[214] Keromytis, Angelos D., Vishal Misra, and Dan Rubenstein. ”SOS: An architecture for mitigating DDoS attacks.” IEEE Journal on selected areas in communications 22, no. 1 (2004): 176-188.

[215] Weiler, Nathalie. ”Honeypots for distributed denial-of-service attacks.” In Proceedings. Eleventh IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises, pp. 109-114. IEEE, 2002.

[216] Verma, Amandeep, Rahul Saha, Gulshan Kumar, and Tai-hoon Kim. ”The Security Perspectives of Vehicular Networks: A Taxonomical Analysis of Attacks and Solutions.” Applied Sciences 11, no. 10 (2021): 4682.

[217] Chapman, D. Brent. ”Network (In) Security Through IP Packet Filtering.” In USENIX Summer, vol. 21. 1992.

[218] Verma, Abhilash. ”Production Honeypots: An Organization’s view.” (2004). Global Information assurance Certification Paper, SANS institute 2004.

[219] Daimi, Kevin, Guillermo Francia, Levent Ertaul, Luis Hernandez Encinas, and Eman El-sheikh, eds. Computer and network security essentials. Springer, 2018.

[220] Kim, Hong-Geun, Dong-Jin Kim, Seong-Je Cho, Moonju Park, and Minkyu Park. ”An efficient visitation algorithm to improve the detection speed of high-interaction client honeypots.” In Proceedings of the 2011 ACM Symposium on Research in Applied Computation, pp. 266-271. 2011.

[221] Ng, Chee Keong, Lei Pan, and Yang Xiang. Honeypot frameworks and their applications: a new framework. Springer Singapore, 2018.

[222] Spitzner, Lance. ”Honeytokens: The other honeypot.” (2003).

[223] Spitzner, Lance. ”Honeypots: Catching the insider threat.” In 19th Annual Computer Security Applications Conference, 2003. Proceedings., pp. 170-179. IEEE, 2003.

[224] Jiang, Xuxian, Dongyan Xu, and Yi-Min Wang. ”Collapsar: A VM-based honeyfarm and reverse honeyfarm architecture for network attack capture and detention.” Journal of parallel and distributed computing 66, no. 9 (2006): 1165-1180.

[225] Le, Anh, Ehab Al-Shaer, and Raouf Boutaba. ”On optimizing load balancing of intrusion detection and prevention systems.” In IEEE INFOCOM Workshops 2008, pp. 1-6. IEEE, 2008.

[226] Keromytis, Angelos D., Vishal Misra, and Dan Rubenstein. ”SOS: An architecture for mitigating DDoS attacks.” IEEE Journal on selected areas in communications 22, no. 1 (2004): 176-188.

[227] Fortunati, Stefano, Fulvio Gini, Maria S. Greco, Alfonso Farina, Antonio Graziano, and Sofia Giompapa. ”An improvement of the state-of-the-art covariance-based methods for statistical anomaly detection algorithms.” Signal, Image and Video Processing 10, no. 4 (2016): 687-694.

[228] Lee, Tsung-Han, Lin-Huang Chang, and Chao-Wei Syu. ”Deep learning enabled intrusion detection and prevention system over SDN networks.” In 2020 IEEE International Conference on Communications Workshops (ICC Workshops), pp. 1-6. IEEE, 2020.

[229] Ali, Amir, and Muhammad Murtaza Yousaf. ”Novel three-tier intrusion detection and prevention system in software defined network.” IEEE Access 8 (2020): 109662-109676.

[230] Rohit, Mehboob Hasan, Sakif Md Fahim, and Abu Hurayra Asif Khan. ”Mitigating and Detecting DDoS attack on IoT Environment.” In 2019 IEEE International Conference on Robotics, Automation, Artificial-intelligence and Internet-of-Things (RAAICON), pp. 5-8. IEEE.

[231] Jamader, Asik Rahaman, Puja Das, and Biswa Ranjan Acharya. ”BcIoT: Blockchain based DDos Prevention Architecture for IoT.” In 2019 International Conference on Intelligent Computing and Control Systems (ICCS), pp. 377-382. IEEE, 2019.

[232] Liu, Zhuotao, Yuan Cao, Min Zhu, and Wei Ge. ”Umbrella: Enabling ISPs to offer readily deployable and privacy-preserving DDoS prevention services.” IEEE Transactions on Information Forensics and Security 14, no. 4 (2018): 1098-1108.

[233] Poongodi, M., V. Vijayakumar, Fadi Al-Turjman, Mounir Hamdi, and Maode Ma. ”Intrusion prevention system for DDoS attack on VANET with reCAPTCHA controller using information based metrics.” IEEE Access 7 (2019): 158481-158491.

[234] Poongodi, M., Mounir Hamdi, Ashutosh Sharma, Maode Ma, and Pradeep Kumar Singh. ”DDoS detection mechanism using trust-based evaluation system in VANET.” IEEE Access 7 (2019): 183532-183544.

[235] Aldaej, Abdulaziz. ”Enhancing cyber security in modern internet of things (iot) using intrusion prevention algorithm for iot (ipai).” IEEE Access (2019).

[236] Rahman, Md Mahmudur, Shanto Roy, and Mohammad Abu Yousuf. ”DDoS Mitigation and Intrusion Prevention in Content Delivery Networks using Distributed Virtual Honeypots.” In 2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT), pp. 1-6. IEEE, 2019.

[237] Ahuja, Nisha, and Gaurav Singal. ”DDoS attack detection & prevention in SDN using OpenFlow statistics.” In 2019 IEEE 9th International Conference on Advanced Computing (IACC), pp. 147-152. IEEE, 2019.

[238] Dao, Nhu-Ngoc, Duc-Nghia Vu, Yunseong Lee, Minho Park, and Sungrae Cho. ”MAEC-X: DDoS prevention leveraging multi-access edge computing.” In 2018 International Conference on Information Networking (ICOIN), pp. 245-248. IEEE, 2018.

[239] Suchitra, M., S. M. Renuka, and Lingaraj K. Sreerekha. ”DDoS Prevention Using D-PID.” In 2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS), pp. 453-457. IEEE, 2018.

[240] Martin, Daniel. ”Early warning of bank failure: A logit regression approach.” Journal of banking & finance 1, no. 3 (1977): 249-276.

[241] Green, Ido, Tzvi Raz, and Moshe Zviran. ”Analysis of active intrusion prevention data for predicting hostile activity in computer networks.” Communications of the ACM 50, no. 4 (2007): 63-68.

[242] Mukkhopadhyay, Arunabha, and G. K. Shukla. ”Mitigating security breaches through insurance: Logit and Probit models for quantifying e-risk.” AMCIS 2009 Proceedings (2009): 767.

[243] Geng, Jinkun, and Ping Luo. ”A novel vulnerability prediction model to predict vulnerability loss based on probit regression.” Wuhan University Journal of Natural Sciences 21, no. 3 (2016): 214-220.

[244] Mukhopadhyay, Arunabha, Samir Chatterjee, Kallol K. Bagchi, Peteer J. Kirs, and Girja K. Shukla. ”Cyber risk assessment and mitigation (CRAM) framework using logit and probit models for cyber insurance.” Information Systems Frontiers 21, no. 5 (2019): 997-1018.

[245] Prakash, PG Om, K. Sasirekha, and Daniel Vistro. ”A DDoS Prevention System Designed Using Machine Learning for Cloud Computing Environment” International Journal of Management (IJM) 11, no. 10 (2020).

[246] Sharma, Kalpit, and Arunabha Mukhopadhyay. ”Cyber risk assessment and mitigation using logit and probit models for DDoS attacks.” (2020).

[247] Verma, Karan, Halabi Hasbullah, and Ashok Kumar. ”Prevention of DoS attacks in VANET.” Wireless personal communications 73, no. 1 (2013): 95-126.

[248] Malhi, Avleen Kaur, and Shalini Batra. ”Genetic-based framework for prevention of masquerade and DDoS attacks in vehicular ad-hocnetworks.” Security and Communication Networks 9, no. 15 (2016): 2612-2626.

[249] Poongodi, M., V. Vijayakumar, Fadi Al-Turjman, Mounir Hamdi, and Maode Ma. ”Intrusion prevention system for DDoS attack on VANET with reCAPTCHA controller using information based metrics.” IEEE Access 7 (2019): 158481-158491.

[250] Prabakeran, S., and T. Sethukarasi. ”Optimal solution for malicious node detection and prevention using hybrid chaotic particle dragonfly swarm algorithm in VANETs.” Wireless Networks 26, no. 8 (2020): 5897-5917.

[251] Al-Mehdhara, Mohammed, and Na Ruan. ”MSOM: Efficient Mechanism for Defense against DDoS Attacks in VANET.” Wireless Communications and Mobile Computing 2021 (2021).

[252] Bajracharya, Brihat.”Detecting DDoS Attacks Using Logistic Regression.” A seminar report, (2020).

[253] Stolfo, Salvatore J., Wei Fan, Wenke Lee, Andreas Prodromidis, and Philip K. Chan. ”Cost-based modeling for fraud and intrusion detection: Results from the JAM project.” In Proceedings DARPA Information Survivability Conference and Exposition. DISCEX’00, vol. 2, pp. 130-144. IEEE, 2000.

[254] Tristan Carrier, Princy Victor, Ali Tekeoglu, Arash Habibi Lashkari,” Detecting Obfuscated Malware using Memory Feature Engineering”, The 8th International Conference on Information Systems Security and Privacy (ICISSP), 2022