Available online at www.garmian.edu.krd

Passer Journal of Basic and Applied Sciences

Passer

Passer 7 (Issue 2) (2025) 953-982

Original Article

Security Challenges and Architectures in SDN-Enabled VANETs: A Comprehensive Survey and Taxonomy

Dana Kareem Hama *[1,2], Foad Salem Mubarek [3], Firas Abdulhameed Abdullatif [4]

[1] Department of Computer Science. College of Computer Science and Information Technology, University of Anbar, 46001, Iraq.
[2]Department of Computer Science, College of Science, University of Garmian, Sulemaniya, Kalar 46021, Kurdistan region, Iraq.
[3] College of Computer Science and Information Technology, University of Anbar, Ramadi, Anbar 31001, Iraq.
[4] Department of Computer Science, College of Education for Pure Sciences (Ibn Al-Haitham), University of Baghdad, Baghdad 10001, Iraq

Received 15 April 2025; revised 29 June 2025;
accepted 09 July 2025; available online 09 September 2025

DOI: 10.24271/PSR.2025. 517062.2078

ABSTRACT

The integration of Software-Defined Networking (SDN) with vehicle Ad Hoc Networks (VANETs) has created a revolutionary framework for overseeing advanced vehicle communications in Intelligent Transportation Systems (ITS). This integration renders Software-Defined Automotive Networks (SDVNs) susceptible to various significant security vulnerabilities, such as Denial-of-Service (DoS) attacks, spoofing, session hijacking, and routing manipulation, due to the centralized architecture of SDN controllers and the elevated mobility within automotive contexts. This study comprehensively analyzes SDN-enabled VANETs (SDVNs), focusing on security frameworks, vulnerabilities, and mitigation strategies. Moreover, the Denial-of-Service (DoS) and routing assaults are recognized as the most disruptive security vulnerabilities, severely compromising network availability, performance, and trust in vehicular communications. The paper delineates the fundamental architectural elements of SDVNs, encompassing centralized, distributed, and hybrid controller models, and examines their operational attributes and design compromises. A detailed taxonomy of over 30 security attacks is presented, classified by origin, goal, and impacted network layer. The survey assesses mitigation strategies, encompassing intelligent and lightweight technologies like SLICOTS, SDNShield, Flood Guard, and LineSwitch. The security posture of SDVNs is evaluated using an advanced CIA3 model that includes Confidentiality, Integrity, Authentication, Availability, and Access Control. The research also emphasizes the relevance of SDVN frameworks in practical vehicle contexts, including emergency communication, collision avoidance, intelligent routing, and smart parking systems. The paper ultimately delineates critical research deficiencies, technical constraints, and prospects for future progress in the secure implementation of adaptive, scalable vehicle networks.

Keywords: VANET, SDVN, Software-defined vehicular networks, Security, Intelligent Transportation Systems, SDN, RSU.

1 Introduction

SDN technology is regarded as a viable approach for mobile communication due to its flexibility, programmability, and network abstraction capabilities. Software-defined networking (SDN) is a novel framework that provides flexible network architecture and management. The SDN paradigm segregates the network architecture into two fundamental planes: the control plane and the data plane. The concept of SDN revolves around segregating control functions (the control plane) from forwarding operations (the data plane). The control plane comprises a centralized controller tasked with all decision-making activities. The data plane refers to the basic forwarding components controlled by a centralized controller. Consequently, the SDN controller, located within the control plane, formulates and oversees the regulations implemented on the forwarding devices, or SDN switches, in the data plane. The two aircraft exchange information utilizing the open-flow protocol[1].

SDN controllers and switches can utilize a standard set of messages to communicate with one another, and the OpenFlow protocol enables this functionality. To create or edit switch rules, refer to these messages. Specifically, the rules govern the SDN controller's predetermined decisions, instructing the SDN switches to perform tasks like packet forwarding and dropping. Essential metrics, network architecture information, and user requirements are also included. A holistic view of the network and simplified management are made possible thanks to SDN's programmability, flexibility, and centralized intelligence services. The SDN controller provides up-to-the-minute network information for various data plane applications and services.

* Corresponding author
E-mail address dana.kareem@gramian.edu.krd (Instructor).
Peer-reviewed under the responsibility of the University of Garmian.

However, there are additional scalability, performance, and security concerns that arise when SDN is integrated with VANETs, even though SDN offers a versatile framework for centralized network management [1,2].

The control plane comprises SDN controllers and forwarding devices interconnected via wireless radio access networks (WRANs) or wired networks. Every controller features an accessible Application Programming Interface (API) that facilitates the development of network infrastructure. A southbound Application Programming Interface (API) connects the centralized controller and forwarding nodes, utilizing OpenFlow as the prevailing protocol. An API that operates in the northbound direction enables administrators to manage the forwarding plane's rules remotely. The SDN architecture prioritizes the forwarding of packets based on data flow regulations, enabling real-time configuration modification and simultaneous administration of essential services such as bandwidth management, security, and access control[3-6].

OpenFlow switches comprise multiple tables and a communication link to the controller, enabling the implementation and integration of additional services and protocols. Standard APIs allow for centralized administration and control of networking devices[7]. This survey identified DoS/DDoS and routing assaults as the most successful and often targeted risks in SDVN setups, owing to their capacity to inundate SDN controllers and distort vehicle routing protocols. To address these problems with VANET, a key strategy is to integrate Software-Defined Networking (SDN) with VANET. An improved user experience was achieved through the dynamic administration of networking resources and the sufficient provision of networking services, made possible by SDN's remarkable qualities, including its existing global topology. To address the current problems with VANET, a key strategy is to integrate Software-Defined Networking (SDN) with VANET. An improved user experience was achieved through the dynamic administration of networking resources and the sufficient provision of networking services, made possible by SDN's remarkable qualities, including its existing global topology. Among other things, these SDN features may meet the complicated needs of VANETs, which include a high data transmission rate, rapid movement, minimal communication delay, a variety of network configurations, and the ability to manage large-scale networks[6-8].

The network is comprised of a collection of programmable switches, commonly referred to as white boxes. The white boxes are connected to one or more SDN controllers using an out-of-band network. The execution of control and management is carried out through a distinct interface. Switches transform into Uncomplicated forwarding devices that adhere to regulations set by the controllers[9]. Figure 1 represents the fundamental architecture of SDVNs. The graphic illustrates the communication between cars equipped with On-Board Units (OBUs) and Roadside Units (RSUs), which are overseen by a centralized Software-Defined Network (SDN) controller. The architecture delineates the control plane (the decision-making layer) from the data plane (packet forwarding), facilitating real-time traffic management, security, and routing policies via programmable interfaces. Components communicate over wireless links utilizing standards such as OpenFlow [10]. SDN-based VANETs offer centralized network control, enhanced administration of vehicular networks, improved security measures, and the capability to enhance communication services. SDN-based VANETs aim to transform the control and management of vehicular networks by leveraging SDN technology, leading to safer and more efficient transportation systems [11-13].

This survey, unlike those in Table 1, examines cutting-edge SDN-based VANET systems with an emphasis on security concerns in vehicular networks. Its notable contributions are enumerated below. It addresses the security challenges inherent to VANET and the reasons for integrating SDN within VANET. Distinctions between our taxonomy and prior surveys: Comprehensive Multi-dimensional Classification. Unlike previous surveys that primarily classify assaults based on the attacker's origin or the impacted network components in isolation, our taxonomy offers an extensive, layered classification. Attacks are methodically categorized according to three dimensions concurrently:

- Origin of Attacker (Internal versus External)
- Intent of Attack (Malicious versus Rational)
- Nature of Activity (Active versus Passive)

Clear Correlation with SDVN Architectural Layers: Our taxonomy explicitly aligns each attack with the control, data, and application planes of the SDVN architecture, effectively highlighting specific weaknesses within each layer. Prior surveys typically lacked this precise and methodical connection. Expanded Scope and Detail, unlike conventional surveys that focus on a narrow range of recognized attacks, our methodology encompasses over 30 attack types, including novel threats such as rule conflicts, switch-based attacks, and counterfeit LLDP packet injections, thereby providing superior granularity and applicability for SDVN environments. By integrating with the Enhanced CIA³ Security Model, we uniquely incorporate our taxonomy into an advanced CIA³ model (Confidentiality, Integrity, Authentication, Availability, and Access Control), offering a comprehensive evaluation framework for assessing and comparing security countermeasures in Software-Defined Vehicular Networks (SDVNs). This broader perspective is often absent in previous studies. This survey aims to address the security challenges of SDN-based VANETs by examining architectural designs, classifying attack vectors, and evaluating mitigation strategies. Here is how the paper is structured:

1. Security Challenges in VANET. SDN aims to enhance vehicular services by integrating transportation, cloud, and edge infrastructures, and wireless communication technologies. By entrusting massive volumes of data to dynamic processing and storage, these services improve throughput while decreasing latency. The dynamic nature of

cars and network topologies necessitates that infrastructure providers provide security measures.

2. This study looks at how SDN has affected VANET security. It then discusses how SDN-based VANETs (SDVNs) impact the network-layer security of vehicular networks and how they are utilized in SDVNs, with a focus on network-layer security requirements.

3. To examine these domains and enhance readability, key elements of SDN-based VANET and vehicular communication technologies are succinctly elucidated, along with their corresponding architectures. The article catalogues SDN-based security services in VANETs, attacks on SDVN, and domain-specific solutions to these problems.

To address security concerns and reach a consensus among SDN controllers in a centralized architecture, a thorough study of security applications in SDVNs is carried out.

Table 1: Summary of SDN-Based Surveys with Security as The Main Concern in Vehicular Networks.

Ref. & Year	**Focus Area**	**Security-Specific Taxonomy**	**SDN Layer Mapping**	**CIA3 Analysis**	**Unique Contributor**
[8],2019	SDVN Architecture, Scalability	General only	×	×	Resource optimization focus
[14],2021	5G-V2X Scenarios	General only	×	×	Emphasis on URLLC & spectrum
[13],2021	SDVN Implementation Challenges	General only	×	×	High-level implementation review
[15],2020	SDN for Future Comms	General only	×	×	SDN methodologies for VANETs, highlighting the challenges and potential solutions.
[16],2022	VANET	General only	×	×	offers a taxonomy of wireless access methods, categorizes services, and suggests solutions for VANET networks
[17],2019	VANET Security Overview	General only	×	×	Basic threat list
[18],2021	VANET	General only	×	×	Privacy-preserving authentication mechanisms in VANETs
[19],2019	Privacy & Trust in VANETs	General only	×	×	Trust-centric discussion
[20],2020	VANET	General only	×	×	security concerns, possible solutions, and their effects on protecting VANETs for road safety and traffic management
[21],2023	VANET	General only	×	×	Trust management inside VANETs
[22],2021	5 G-enhanced VANET Security	General only	×	×	Network infrastructure focus
[23],2021	VANET	General only	×	×	security and privacy concerns in VANETs by categorizing identity-centric security and privacy frameworks
[24],2022	Trust-based SDVN Framework	General only	×	×	Deep RL for trust management
[25],2019	VANET	General only	×	×	in-vehicle communications by comprehensively analyzing security threats in VANETs
This Survey	SDVN	Multi-dimensional	Control, Data, App	Full CIA3 Model	Full-stack, security-centric taxonomy with real-world relevance

Unlike previous surveys, this paper goes beyond listing existing attacks by offering a layered attack taxonomy that categorizes threats by attacker origin, intent, and activity type. Moreover, it compares SDVN-specific security countermeasures, highlighting the design's specific strengths and weaknesses. Finally, Figure 2 visually depicts the study's organization, distinctly delineating the sections that constitute the research.

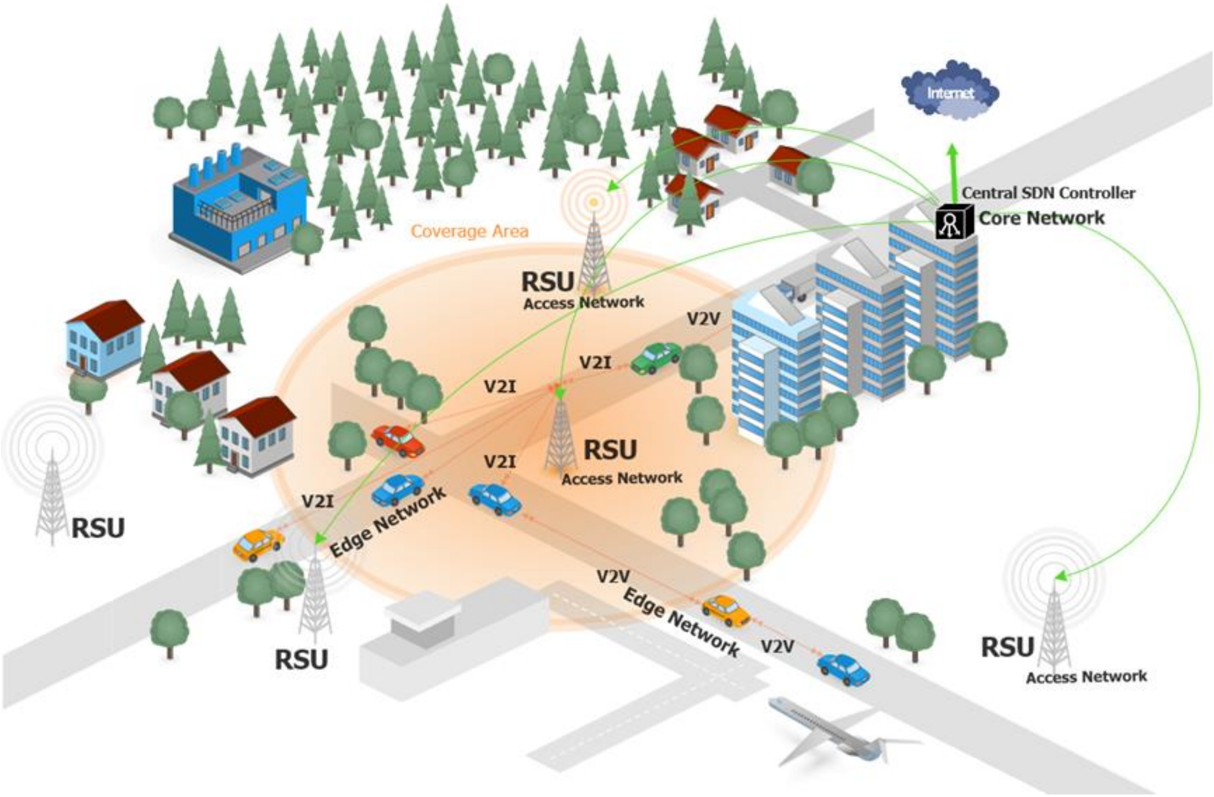

Figure 1: SDN-based VANET Network Architecture.

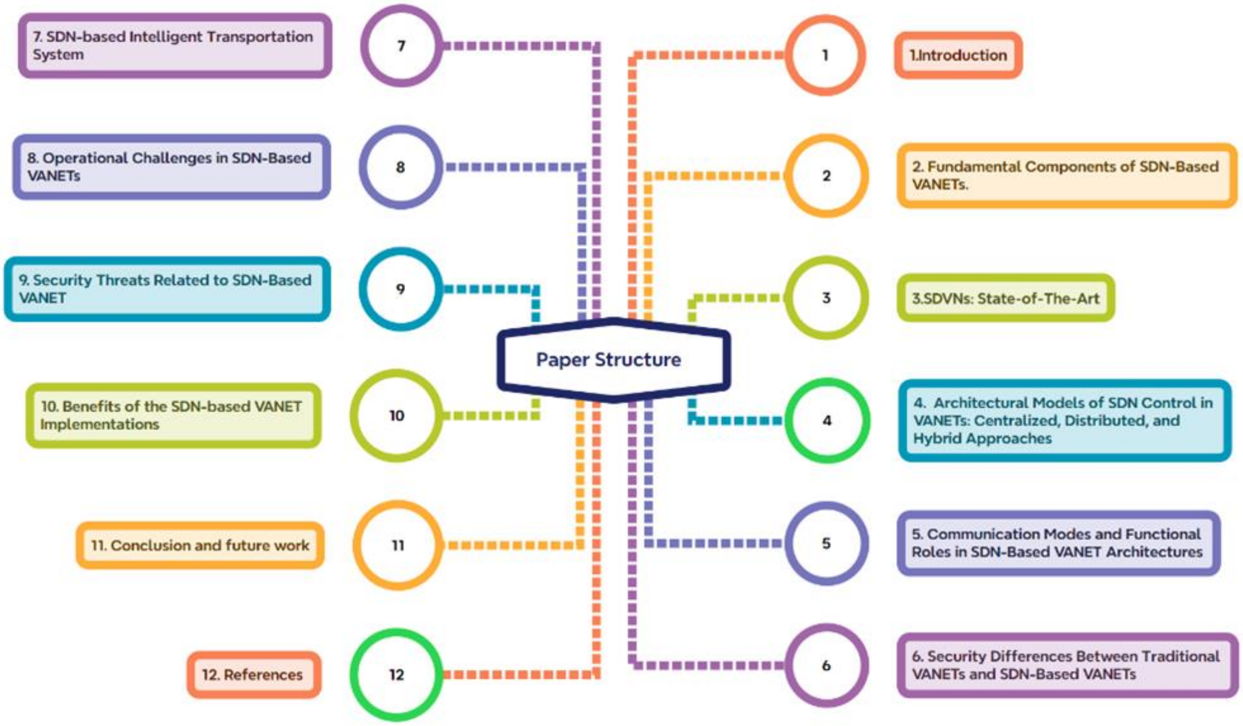
Paper Structure
1
1.Introduction
2
2. Fundamental Components of SDN-Based VANETs.
3
3.SDVNs: State-of-The-Art
4
4. Architectural Models of SDN Control in VANETs: Centralized, Distributed, and Hybrid Approaches
5
5. Communication Modes and Functional Roles in SDN-Based VANET Architectures
6
6. Security Differences Between Traditional VANETs and SDN-Based VANETs
7
7. SDN-based Intelligent Transportation System
8
8. Operational Challenges in SDN-Based VANETs
9
9. Security Threats Related to SDN-Based VANET
10
10. Benefits of the SDN-based VANET Implementations
11
11. Conclusion and future work
12
12. References

Figure 2: Paper Organization.

2 Fundamental Components of SDN-Based VANETs

The SDN wireless node, SDN controllers, and RSU are the three main parts of this system. Figure 3 illustrates the internal architecture of a vehicle within an SDVN system. It demonstrates how On-Board Units (OBUs) facilitate communication between cars and other network components. OBUs manage network connectivity, whereas RSUs facilitate data transmission and user engagement. These components interface with RSUs, which subsequently connect to the SDN controller. This figure illustrates the vehicle's role as both a data source and a network member within the SDVN ecosystem. The SDN controller is utilized to regulate system-wide network behavior. In most cases, it is connected to the RSU. Vehicles that are part of an SDN and interact with one another in a VANET are referred to as nodes. Antenna Units (AU) and On-Board Units (OBU) are part of its equipment. All of them are communicating with one another. An AU's capabilities extend to those of a digital assistant or personal device. Using the OBU device, the AU is responsible for all network connectivity and uses it to communicate across the network. An SDN, or roadside unit, is a permanently installed device fastened to a roadside or parking lot surface. The RSU device is connected to a network, allowing vehicles to communicate with the SDN controller. When an application needs hosting, the RSU is the one to call. The OBU utilizes these services. Many AUs in different vehicles can create online connections thanks to the RSU's internet connectivity[10],[26,27].

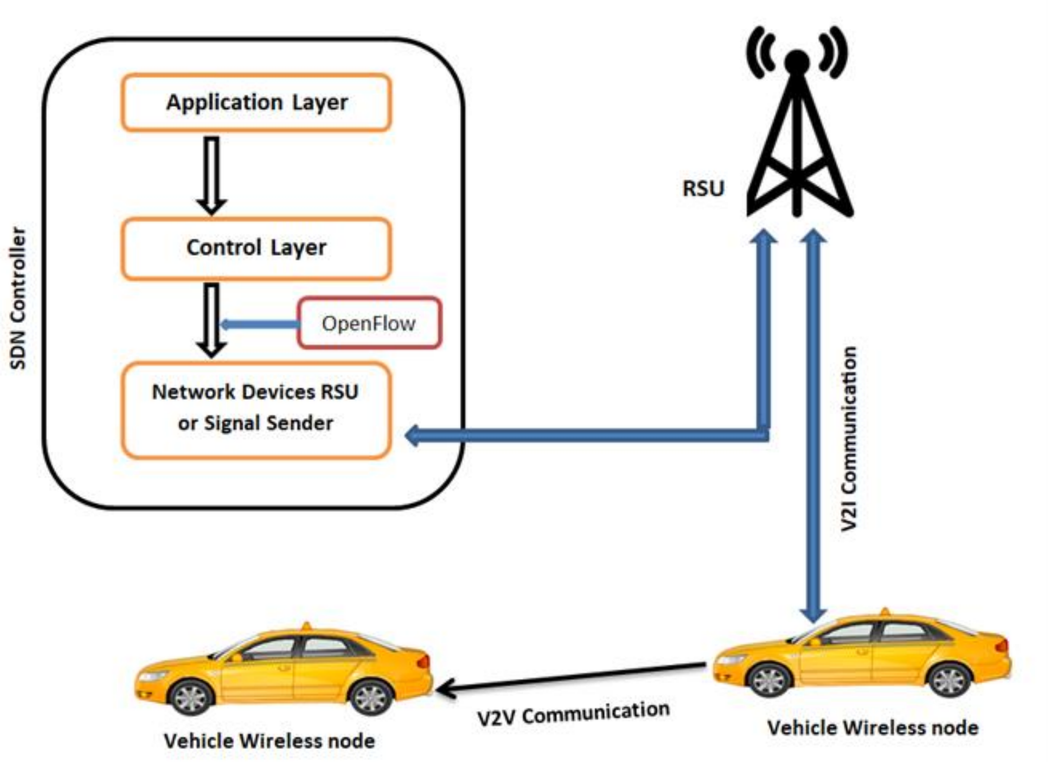

Figure 3: The core concept of SDN-based Vehicle [10].

The three-level model depicted in Figure 4 outlines the Software-Defined Networking architecture utilized in VANET, comprising the Application Plane, Control Plane, and Data Plane. The Application Plane has sophisticated programs for traffic management, intrusion detection, and access control. The Control Plane comprises SDN controllers that execute routing and policy determinations. The Data Plane comprises switches and routers that transmit data packets according to the controller's directives. APIs (northbound and southbound) facilitate communication across layers, ensuring the adaptable and real-time management of the vehicle network. Automated and effective management of network infrastructure is the goal of these levels. The engineering discipline aims to integrate the control and information components in structured devices, such as routers and switches. While data packet transmission is the responsibility of the data plane, decision-making regarding the traffic that flows across all network devices is the responsibility of the control plane. In most cases, there are three levels to a programming system's architecture: control, application, and infrastructure [5],[16].

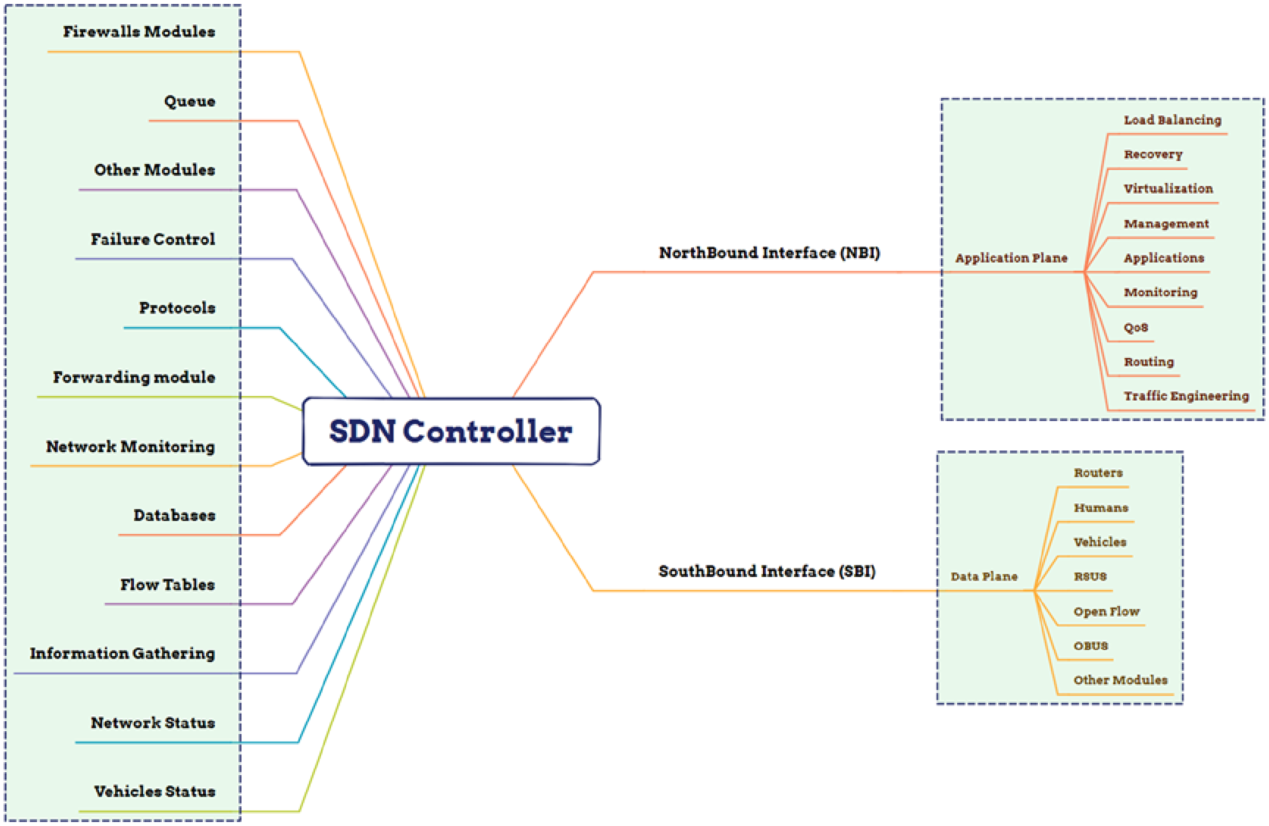

Figure 4: SDN Component.

2. 1 Data Plane

The data plane, often referred to as the foundational plane, consists of system devices, including routers, switches, and gateways, that are responsible for facilitating the movement of all user data flowing over the network. These system gadgets do not expand and have a predetermined and unchanging use, yet they are governed by several criteria set by the control layer. By configuring the system supervisor's settings, the same equipment may be used either as a switch or as a firewall [5],[10,6].

2. 2 Control Plane

The control plane is responsible for centralizing the control of the entire data flow that circulates through the data or infrastructure layer. This encompasses the methods of transmitting or diverting information, as well as flow tables, and the overall perspective of the entire system, aided by a software-defined networking SDN controller. The Application Programming Interface (API) southbound interface (SBI), such as the OpenFlow protocol, allows controllers like Open Sunlight or VMware NSX (a network virtualization and security platform) to transmit the arrangement of methods and configurations to all the devices that comprise the data plane. The APIs play a crucial role in accurately dividing the components of data and the control systems. There are also northbound interface (NBI) APIs, such as the RESTful API or the software-defined mobile networks (SDMN) API, that facilitate dynamic flow programming. They play a vital role in bridging the gap between global application approaches and system strategies[5,6].

A third-party organization known as the SDN controller oversees all network operations. Advanced orchestration and provisioning technologies allow for successful administration when a centralized controller is used, which has broad awareness of the entire network. The optimization of routing and switching equipment is made easier, and the potential for building services and applications on the network abstraction layer is expanded. In addition, it follows the network-wide directives for practical forwarding algorithms. Through the widely recognized and acknowledged OpenFlow protocol, the controllers communicate with the controller agents in the physical network devices[10,28].

2. 3 Application Plane

The application layer is a system's ultimate layer, where several application programs are created to interact and collaborate with other designs. The NBI APIs enable the rapid completion of this layer, offering a distinct perspective of the system encompassing device distribution and assessments of system performance. For the successful implementation of these advanced programs, it is essential to use networks that support open-source development, prioritize security and encryption, and demonstrate adaptability[5],[6,10].

3 SDVNs: State-of-the-Art

Numerous improvements have marked the evolution of SDVNs. Various technological aspects and design components have improved throughout time. This section provides a few examples of relevant literary works and contemporary technological developments to emphasize the current state of the art. Functioning as a potential technology, SDVN can enhance the efficiency of modern vehicle networks[29].Hammad Shafiq et al.. investigate the incorporation of Software Defined Networking (SDN) within Vehicular Ad Hoc Networks (VANETs) to improve safety and efficiency in transportation. The text addresses the difficulties of conventional VANETs, encompassing security vulnerabilities and communication problems, and proposes SDN as a remedy for enhanced network administration[10].With SDN, the controller adds another attack vector—the link between the controller and the switch—to the network. To get into the controller and its network, attackers use various techniques. Four major categories of threats and possible attacks on SDVNs are covered in this section[30].

In SDN architecture, a topology discovery technique that is both efficient and accurate is required so that the controller can enable effective network administration. To set up a network topology based on software-defined networking, it is necessary to discover hosts, links, and connections between switches. By default, most SDN controller platforms use the OpenFlow Topology Discovery Protocol (OFDPP) [30]. Many ad hoc routing methods have been developed to address VANET problems; nevertheless, these methods manage a dynamic network topology and use communication bandwidth-depleting HELLO messages. Implementing SDN in VANETs can detect malfunctioning switches and promote optimal routing choices, thus resolving this issue. An artificial neural network is used by SDN-based routing techniques and architectures to learn and predict the arrival rates of vehicles [31,33].

The goal of a denial-of-service (DoS) attack is to prevent authorized users and services from accessing a resource, such as a website, application, or server. You can make this attack work by flooding the target resource with traffic from different network protocols. Poison ping, teardrop, UDP/SYN flood, and SYN flood are some of the many methods that can be used to launch denial-of-service attacks. The severity of these attacks is greatly amplified. The controller's resources, bandwidth, and processing capability are depleted when switches flood it with an excessive number of packets [30].

Network-based applications within the SDVN paradigm possess exploitable weaknesses, necessitating security audits and penetration testing. These vulnerabilities may result in unauthorized access controls, malicious packet flooding, firmware updates, zero-day exploits, and data breaches. APIs utilized for communication with controllers are susceptible to vulnerabilities. Therefore, standardized Northbound Interface (NBI) and OpenFlow (OF) APIs should be evaluated for security measures [34].Table 2 presents the potential countermeasures and security attacks in SDVN. The analysis indicates that DoS and routing assaults are the primary weaknesses addressed in SDVN, frequently countered by artificial intelligence and flow-rule optimization techniques.

Table 2: Security Attacks and Proposed Countermeasures in SDVN.

Author &year	Attack	Result of Attack	Affected Layers	Proposed Countermeasures
[6],2019	DDoS, impersonation, jamming, and forgery attacks	Risks to availability, confidentiality, authentication, and data integrity.	Control plane, Data plane, Application plane	Control Plane Resource Utilization: Executing packet migration and data plane caching to optimize flow rule management. Network Topology Poisoning: Employing Topo Guard and SPHINX for surveillance and verifying topology modifications. Utilizing machine learning methodologies for detection and implementing Flood guard for packet movement. Rule Conflicts: Incorporating Fort NOX to identify and implement security protocols.
[35],2021	DoS Attack	network unavailability, packet drop, failure of the entire network, and communication channel flooding	Control plane, Management plane	The Flood Defender system effectively identified and alleviated DoS attacks, minimizing their effect on the controller and network resources. The application-based solution alleviated the burden on the OpenFlow switches by regulating flows and establishing timeouts according to traffic statistics. The Line Switch system employed switch intelligence to detect DoS attacks at the data layer.
[36],2020	Routing Attack	An attack compromises the efficacy and security of data transmission.	infrastructure layer	The suggested solution, Efficient and Secure Trusted Routing Mechanism (ESTRM), aims to deliver a secure and optimal pathway for data transfer in software-defined networks.
[37],2020	Routing Attack	The attacks include Sequence and Data-	Control plane	DELTA is a proposed SDN penetration tool designed to replicate established attack scenarios and identify

		Forge, Statistics Payload Manipulation, Echo Reply Payload Manipulation, and Service Unregistration, targeting SDN components and control flows.		previously unknown security vulnerabilities within SDN infrastructures. It employs a fuzzing module to randomize message input values and identify vulnerabilities.
[38],2021	Network Topology Poisoning	The attack involves assessing the information risk linked to SDN or SDN-based architecture.	Control plane, Data plane, Application plane	The proposed algorithm employs Multi-Criteria Decision-Making (MCDM) methods to assess SDN attributes and vulnerabilities, detect system hazards, and allow network managers to implement suitable remedies for improved security and sustainability.
[39],2020	DoS Attack	When organizations move to cloud-based infrastructure, protecting SDN interfaces from assaults is critical to avoiding network collapse. Maintaining the security and dependability of SDN networks in the ever-changing IT world requires a deep understanding of vulnerabilities and effective mitigation strategies, such as link flooding attacks.	Control plane, Data plane	Approaches based on principles, traffic engineering, and link monitoring are all proposed solutions to the problem of link flooding attacks in software-defined networking. These efforts aim to reduce the impact of LFAs on the SDN environment's application, control, and data planes. To counter these attacks, it is recommended to use detection and mitigation methods based on machine learning and pattern matching.
[33],2021	Routing Attack	SDVN must be scalable to accommodate the increasing volume of cars; the attack affects the scalability of SDVN	Control plane, Data plane, Application plane	SDVN's hierarchical control design enhances scalability by distributing control decisions and optimizing network performance. The hybrid control architecture integrates centralized and distributed control to maximize load distribution and improve network efficiency.
[40],2021	DoS Attack	delay, expense, load, and capacity of the controller	Control plane	The controller placement problem in SDN is analyzed by considering multiple limitations, including latency, cost, load, and capacity. Proposed solutions encompass the Pareto Optimal Controller (POCO), QoS -Guaranteed Controller Placement, and LiDy+ algorithms. These technologies seek to enhance controller positioning for effective traffic management, dependability, and cost-efficiency.
[41],2021	link flooding attack (LFA) under the framework of DDoS attacks	low-latency transmission, traffic management, congestion control, load balancing, and flow table management.	Data plane	Controller-based load balancing is a proposed remedy for imbalanced network operations in Software-Defined Networking (SDN). It allocates the control plane across many controllers. Essential elements encompass controller positioning and switch transition. Optimization techniques such as multi-objective combinatorial optimization and game-theoretic mapping facilitate

				effective load balancing, enhancing network performance, and diminishing latency.
[42],2021	DDoS Attack	The depletion of control plane bandwidth and the switch's flow table, resulting in network congestion and packet loss	Control plane, Data plane	Diverse techniques, including rate-limiting packets to the controller, mitigating buffer overflow, and hashing incoming packet headers, have been suggested to counter these assaults. Statistical methodologies include entropy-based detection, machine learning algorithms for intrusion detection, and the implementation of Network Function Virtualization, Honeynets, and Moving Target Defense for mitigation.
[43],2021	RSU compromisation attack	This assault entails an assailant seizing the RSU and appropriating information, hence undermining the security system.	Data plane	Establishing a unified session key for secure communication, safeguarding against RSU compromise via authentication protocols, and guaranteeing the absence of security vulnerabilities.
[44],2020	DoS/DDoS attacks	Degrade network performance, discard genuine messages, and render controller functionalities inaccessible to authorized users.	Control plane	Flow Visor and Virtual Source Address Validation Edge (VAVE) can help alleviate these security vulnerabilities in Software-Defined Networking (SDN) systems.
[45],2019	malicious applications, DoS/DDoS attacks, and flow rules conflict	Inconsistencies in flow rules, absence of application authorization, and deficiencies in encryption measures	Control plane, Data plane, Application plane	Authorized authentication modules, application isolation, DoS/DDoS mitigation, multi-controller deployment, and flow rule consistency verification
[46],2023	DDoS attacks, ARP spoofing attacks, and flow rule conflicts	Weaknesses and security issues in SDN architecture	Control plane	Safety solution for TCP SYN flooding, SLICOTS for alleviating TCP SYN flooding at the control plane, SDN-based CDNI network architecture for identifying faked IP addresses, deep learning-enabled autonomous detection, and mitigation of DDoS attacks.
[47],2020	injection attacks, man-in-the-middle attacks, DDoS, and flood attacks	Disarray in the network topology, resulting in irregular network routing or switching, prevents the SDN controller from accurately determining routing paths.	Control plane	The proposed Correlation-based Topology Anomaly Detection (CTAD) mechanism uses Spearman's rank correlation to identify anomalous LLDP packets and thwart attackers from creating counterfeit links. This approach incorporates dynamic authentication keys and counting systems to prevent attacks, safeguarding the network's security and stability.
[48],2016	Replay attacks and jamming attacks	In this scenario, threats significantly influence drivers' behavior. Jeopardized the safety, privacy, and quality of life for drivers	Control plane, Data plane	Implementing globally synchronized time and nonce values can thwart replay attempts, but the dynamic selection of communication channels can alleviate the effects of jamming assaults.
[49],2020	Network Harvesting attack, spoofing attacks	Network Harvesting (NH) attacks enable an attacker to illicitly acquire network access credentials	Control plane, Data plane, Application plane	The execution of a detection system called RS Detector and a defense mechanism known as Spoof Defender. RS Detector employs machine learning to identify rogue switches in real-time, whereas Spoof Defender

		from victims who are unknown to them.		safeguards networks against spoofing assaults with global control.
[50],2021	Critical targeted attacks	Malicious cyberattacks happen due to poor network security, compromising connectivity and continuity.	Control plane	An optimization strategy for positioning controllers in SDN networks to enhance resilience against significant targeted assaults.
[51],2022	DDoS attacks	decreased service quality, operational disruptions across the entire network, and scalability issues	Control plane	SDNShield presents a defensive mechanism against DDoS attacks utilizing NFV technologies. It employs a three-tier overload management mechanism that statistically identifies legitimate flows, conducts TCP handshake verification, and rectifies misclassified lawful flows.
[52],2022	DoS attacks	The breakdown of the SDN control channel results in significant network anomalies, including routing blackholes, flow table resets, and widespread DoS attacks.	Control plane	A unique technique termed the adversarial path was proposed to detect paths targeted by attacks. Cross Guard is an efficient technology that safeguards the SDN control channel efficiently and swiftly identifies the assault flow.
[53],2020	Potential for long-range attacks	Propagation of misinformation, denial-of-service attacks, and unauthorized access to essential vehicle systems.	Control plane, Data plane, Application plane	Establishing centralized control for enhanced network management, leveraging machine learning for anomaly detection, and utilizing network slicing for comprehensive isolation.

4 Architectural Models of SDN Control in VANETs: Centralized, Distributed, and Hybrid Approaches

The SDN-based VANET systems can support all three SDN control models: unified and appropriate. The three models possess a clearly defined basis and specific requirements. Below, you can find the explanations for these three models[5,10].

4. 1 Centralized Controller Model

This model has a singular controller employed to oversee the entire system. This indicates that the SDN is augmented by the OpenFlow rules standard, which controllers utilize and incorporate to manage and administer the whole system. The controllers use a device directly linked to the network to identify the system's vulnerabilities and attacks. These devices convey data to the controller in a sanctioned and related manner. Utilizing a single controller facilitates the management of the complete network's operation. Nevertheless, a singular controller possesses specific constraints. Figure 5 illustrates the centralized model [5,10].

4. 2 Distributed Controller Model

The SDN-based model addresses individual controller failures and limitations by allocating load across multiple controllers and multi-core systems. This model improves responsiveness, speed, and efficiency in sizable global network regions. However, challenges include creating a logical control plane and data plane mapping, understanding the entire system, and gaining broader access. Local computations are often used for coordination, but synchronization and a comprehensive overview are crucial for successful implementation. Figure 6 depicts the distributed controller model [5,10].

4. 3 Hybrid Controller Architecture

A hybrid control architecture is a good option for overcoming problems with both distributed and centralized SDN controllers. Efficient resource utilization, increased network performance, enhanced system security, and quick updates without adjustments to current settings are all benefits of combining centralized and distributed SDN controller systems. It borrows the distributed controller's data transmission architecture and applies a logical framework identical to the main controllers. Figure 7 shows the model of the distributed controller[5,10].

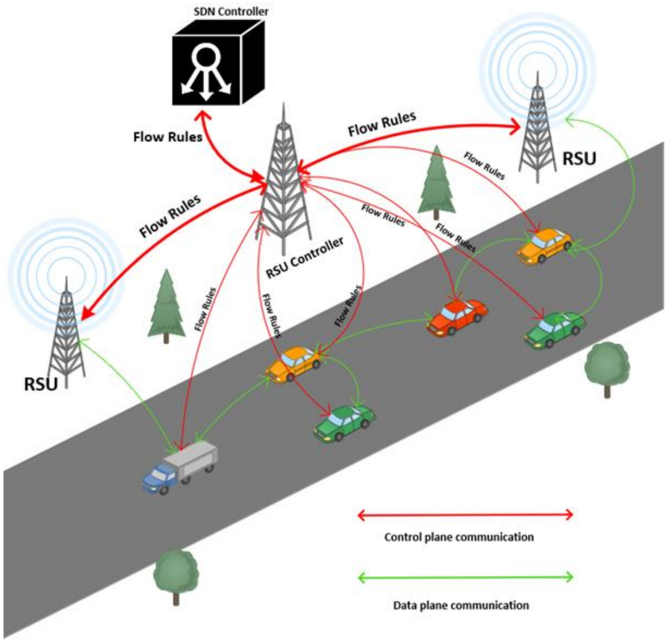

Figure 5: Centralized model of SDN controller.

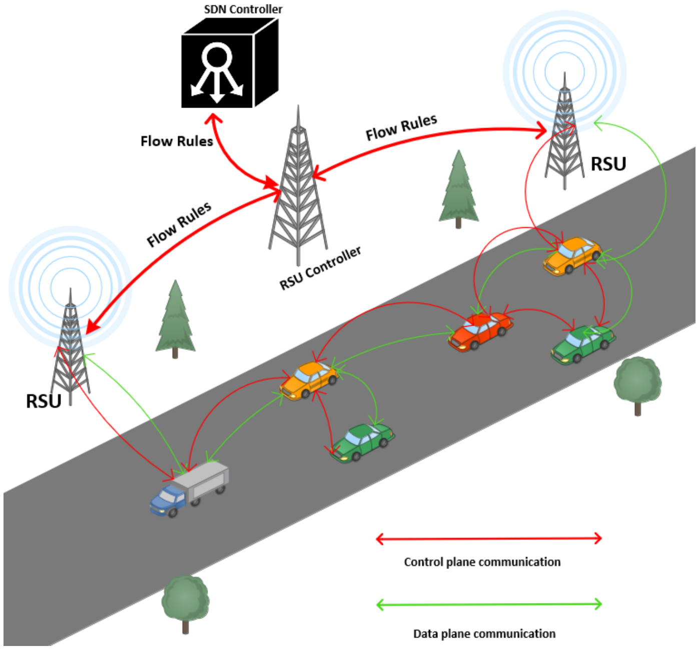

Figure 6: Distributed model of SDN controller.

5 Communication Modes and Functional Roles in SDN-Based VANET Architectures

Three categories define communications in VANETs: (i) intra-vehicle communication, (ii) vehicle-to-vehicle (V2V) communication, and (iii) vehicle-to-infrastructure (V2I) communication. Intra-vehicle communication is the connection of sensors and equipment housed inside the vehicle. Vehicles with intra-vehicle communication have different Electronic Control Units (oecus), sensors, and actuators. Local interconnect networks, controller area networks, media-oriented system transport, Ethernet, and power line communications are among the intra-vehicle communication systems [12,30].

With vehicle-to-vehicle (V2V) communication, a single vehicle can establish one-hop contact with another vehicle. Cars use a routing system to pass messages from one vehicle to another until they reach the destination vehicle if a direct link is unavailable. Innovations in safety features and entertainment options like infotainment and online gaming services can be made possible by vehicle-to-vehicle connections. The evolution of various routing protocols has made many VANET applications possible. Protocols for routing include those for broadcasting, route-discovery, position-based, and clustering-based data transmissions [12,30].

Applications such as video streaming and advertisement distribution are made possible by vehicle-to-infrastructure (V2I) communication, which entails a vehicle communicating with a Roadside Unit (RSU). The RSU infrastructure receives parameters from vehicles, such as their location, speed, and orientation, through predetermined messages. After collecting this data, the RSU will process it and provide the required services, such as location data, video/multimedia streaming, or location-based advertisement distribution. V2X communications, which include V2V, V2I, and vehicle-to-pedestrian interactions, are defined by the Third Generation Partnership Project (3GPP). Access to the Internet while driving, remote diagnostics, entertainment services, traffic optimization, cooperative crash alerts, and many other applications are made possible by vehicle-to-everything network (V2X) connections. Figure 8 shows the communication in a VANET based on software-defined networking [12,30].

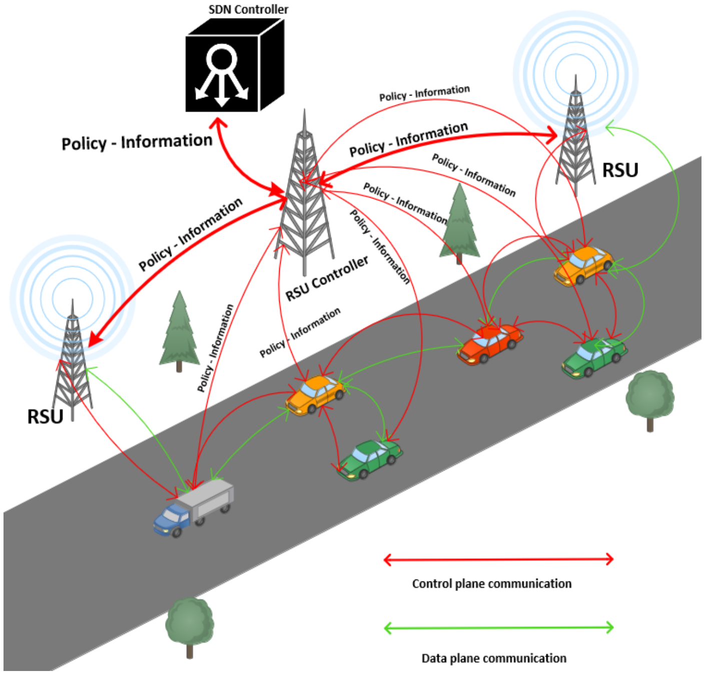

Figure 7: Hybrid model of SDN controller.

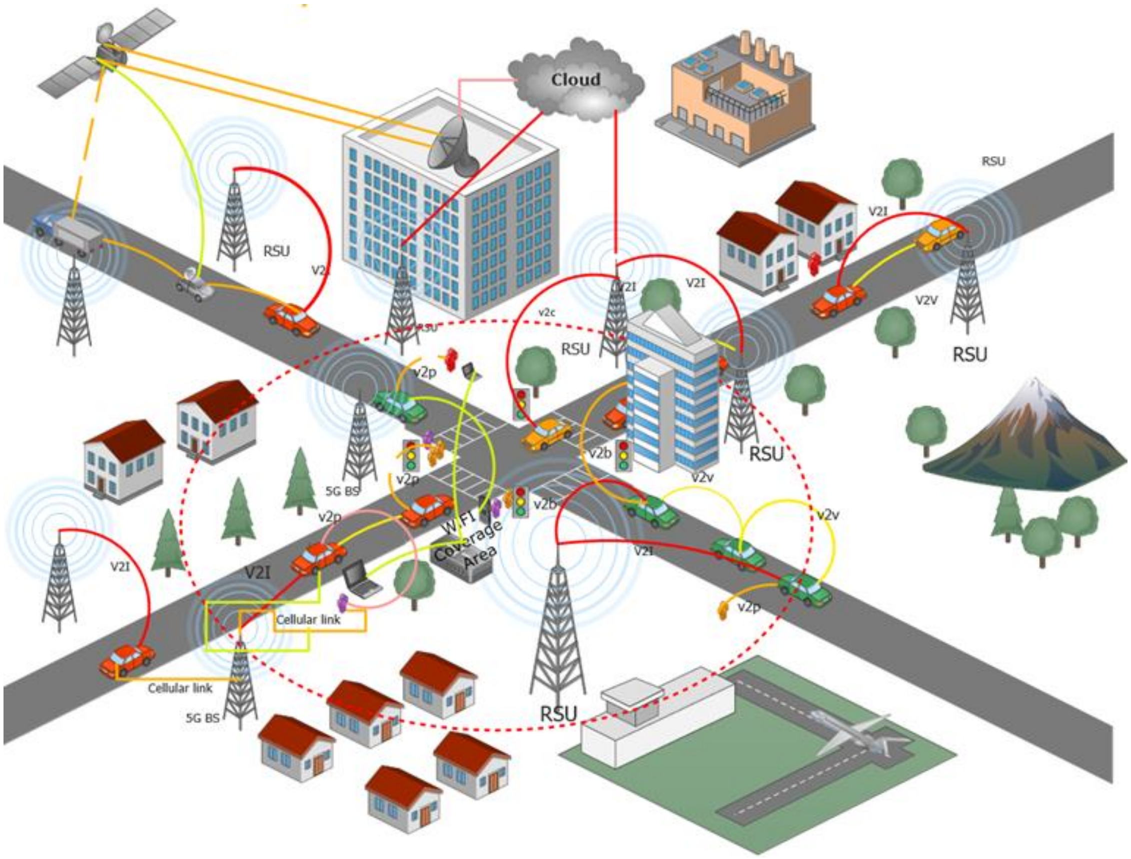

Figure 8: SDN-based VANET Communication Infrastructure.

6 Security Differences Between Traditional VANETs and SDN-Based VANETs

Traditional VANETs operate on a decentralized design, wherein vehicles independently determine routing and forwarding decisions, frequently utilizing established routing protocols such as AODV or DSR. Conversely, SDVNs separate the control plane from the data plane, facilitating centralized management and dynamic programmability with SDN controllers [2], [6,10].

From a security perspective, the principal distinctions encompass:

- **Attack Surface:** Traditional VANETs are susceptible to routing attacks, impersonation, and jamming owing to the absence of centralized oversight. SDVNs, albeit superior in centralized detection, provide new vulnerabilities like controller-targeted DDoS attacks and flow-table saturation [6].

- **Intrusion Detection:** In conventional VANETs, Intrusion Detection Systems are deployed in a dispersed and reactive manner. SDVNs facilitate network-wide IDS with comprehensive visibility, allowing for expedited anomaly detection and reaction via controller logic [10].

- **Rule Management and Policy Enforcement:** Conventional networks lack precise control policies. Software-Defined Networks (SDNs) can dynamically modify flow rules, implement access controls, and isolate malicious nodes in real time [2].

- **Trust and Authentication:** Authentication on conventional VANETs typically depends on localized procedures, such as Public Key Infrastructure (PKI) or group signatures. In SDVNs, centralized trust models and federated identities may be employed, albeit they augment reliance on the controller [91].

- **Resilience and Recovery:** Conventional networks depend on re-routing and exhibit sluggish adaptability. SDVNs provide programmable self-healing capabilities; however, the controller constitutes a single point of failure until distributed or hybrid models are implemented [92].

7 SDN-based Intelligent Transportation System

The Internet has led to a rise in devices connecting to the Web, integrating intelligent transportation systems (ITS) with SDNs to achieve many benefits and deal with other technologies, as illustrated in Figure 9. SDN has simplified traffic management, enhancing traffic flow and environmental conditions on roadways, which aims to improve highway transportation and security. However, existing systems are characterized by sluggishness and high costs, requiring bulky devices on highways, including roadside units (RSUs) and onboard units

(OBUs). Research networks have prioritized the quest for efficient and economical solutions for ITS. This innovative approach revolutionizes systems administration by separating the control layer from the data plane, allowing for efficient management of system devices and services through low-level utility planning[5,54].

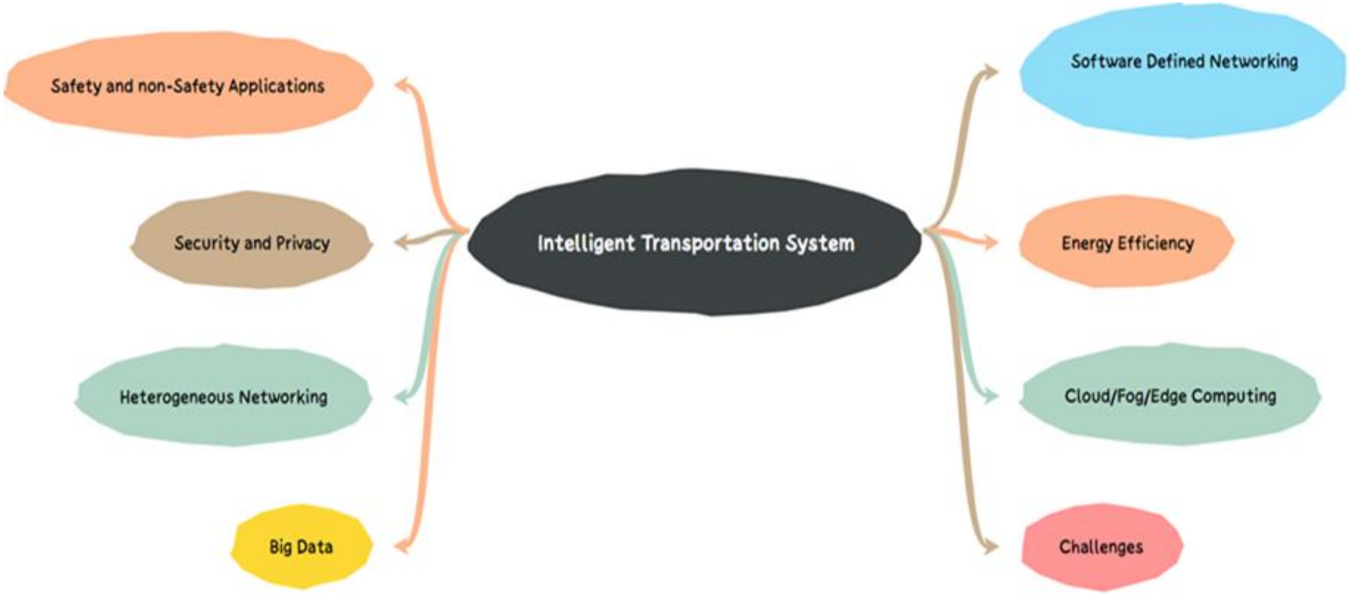

Figure 9: Integration of the SDNs with intelligent transportation systems and other technologies.

8 Operational Challenges in SDN-Based VANETs

Even if it is rapidly evolving, there are many additional challenges concerning the base's efficacy, adaptability, and consistent quality (acceptability). These challenges may determine the future course of the base's development[5]. While SDN-based VANETs do have their advantages, they also present several challenges, some of which are detailed below [13,18].

8. 1 Mobility and Topology Management

The dynamic nature of vehicles' surroundings might impede the implementation of reputation-based strategies. Rating various automobiles based on consistent report quality is challenging due to the limited availability of data for individual vehicles, which hinders the ability to make informed decisions. Conventions necessitate a linkage between the sender and receiver. A revised portability model is required to provide data on distinct vehicular actions, including velocity, credibility, and conveyance[5,27].

8. 1. 1 Highly Dynamic and Distributed Behavior of Vehicles

SDN-based VANETs face challenges due to vehicle-dispersed conduct, causing broad overhead, security delays, and unhealthy programs. To address these issues, it's crucial to distribute parcels based on geological conditions and plan intelligent confinement and direction prediction components [5,27].

8. 1. 2 Mobility-Aware Edge Caching

SDN-based VANETs must optimize edge resource allocation for individual vehicle needs, minimizing delays. Methods include retaining duplicates, probabilistic reserving, dormancy-aware reserving, congestion-aware storage, and Pop Cache. Reserve removal strategies are crucial for new sections and adhering to first-in, last-out, least-used, and random layouts. Assessing substance accessibility is challenging due to rapid velocities and unpredictable drug consumption patterns [5,27].

8. 1. 3 Management of Rapidly Changing SDN-based VANETs

Hubs' flexibility allows for quick modifications in SDN-based VANETs, but controllers and RSUs struggle with controlling cars and facilitating communication. Disrupted connections are more likely due to DSRC or WAVE's inability to handle speed variations [5,27].

8. 2 Switching, Routing, and Forwarding

When it comes to directing and sending, several questions arise regarding the transfer of SDN servers and their services from the source to the destination by the vehicle's movement[5].

a. **SDN Server Switching**. Vehicles often make decisions on their optimal course of action within a short period, as they constantly travel at a high speed. It is difficult to predict which cars will get services from BS or SDN servers based on traffic and public transit data that analyze vehicle movement patterns to anticipate their location in the future[5].

b. **Content Caching.** SDN-based VANETs can improve content storage by pre-fetching and efficient reserving, enabling vehicles to save and promote unspecified compounds. However, there are deficiencies in optimizing vehicle material's temporary and spatial capacity, as it stores substances outside the spatial dimension. Although vehicle-to-vehicle communication has the potential to enhance the network's capacity for content caching, it is currently unable to provide a reliable and high-speed data management system for vehicles due to the complex and unpredictable network topologies and challenging channel conditions[5].

8. 3 RSU Deployment

Installing sufficient network components significantly improves system performance but requires careful selection and strategic deployment. SDN servers and SDN-RSUs are strategically placed to maximize efficiency and manage resources. SDN servers distribute traffic packets efficiently, reducing latency and speeding up communication with other cars. Moreover, additional servers are deployed in crowded areas to address traffic distribution variations [5],[27,13],[55,56].

8. 4 VANETs Connectivity

Heterogeneous vehicle systems administration ensures reliable, standardized system availability for various cars, safety applications, software, and passengers. Advanced radio communication technologies link cars and their clients, guaranteeing minimal delay, fast data transfer, improved information rates, and cost-effectiveness. However, managing the diversity of radio access technologies is challenging, requiring careful decisions on vertical handovers. Automobiles must use unique radio access technologies to traverse geographical areas or topographical regions in distributed vehicular systems administration. Successful vertical handovers involve assessing the need, choosing the optimal network, and deciding the right time [5],[12,13].

8. 5 Heterogeneous Multi-Hop Routing

Vehicles transmit data through multi-hop V2V communication or roadside units. However, short or unattainable links can cause communication failures and system asset waste. Reliable and economical multi-jump steering is crucial in VANET environments, restoring system availability in cases where primary radio access is unavailable [5],[13,27].

8. 6 Broadcast Storm Mitigation

The advancement of onboard sensors and edge communication platforms allows cars to transmit essential messages, such as emergency vehicle warnings and collision alerts, to pedestrians and other vehicles. However, this can cause network congestion, causing traffic to use up resources. Additionally, temper communication can be triggered by malicious entities, requiring network security to prevent infiltration and recovery mechanisms[5,57].

8. 7 Security and Privacy Considerations

SDN-based VANETs rely on an SDN-based controller to manage resources and regulate network services. Protecting these controllers from cyberattacks is crucial, as malicious data dissemination can lead to accidents. DoS attacks can disable controller functions, potentially compromising internal assaults. Ensuring controller security is a central decision-making component in SDN-based VANETs. A thorough investigation of potential security risks is necessary before implementing hybrid systems using SDN-based VANETs[12],[13,27].

8. 8 Fault Tolerance

Failure mitigation in SDN-IoT may be achieved by utilizing backup offloading lines. Furthermore, micro base stations or central clouds with broader coverage can facilitate fault tolerance to provide uninterrupted edge services. Ensuring suitable Quality of Service and energy efficiency for backup connections while sustaining protective clouds for both single-user and multi-user edge computing is a significant problem[13,27].

9 Security Threats Related to SDN-Based VANET

This section describes the problems and security dangers associated with SDN-based VANETS. Security problems must be considered throughout the system's design. Initially, an overview of the various categories of assailants is provided [10]. SDN enables the separation of network security functions from vendor hardware, providing the ability to perform flexible security management operations without changing the hardware. The system offers reactive and proactive security monitoring, analysis, and policy execution. The centralized architecture of SDN enables immediate detection of threats, analysis of network status and flow, changes to policies, and alteration of network flow[58].

Ensuring security in SDN systems is crucial due to the potential for unauthorized dissemination of erroneous information. The central decision-making point should be securely protected, and a comprehensive approach is recommended to preserve SDN systems. Tightly interconnected SDN levels can facilitate risk propagation across different layers, so APIs connecting different levels should be strengthened and standardized. The adaptability and responsiveness of vehicles in the lower information plane increase risks of SDN to control and application levels, necessitating continuous verification. Challenges also impact the security and safety of connected and autonomous vehicles (CAVs), as failures in one sub-system could spread to other sub-systems, hindering recovery. Digital threats and AI vulnerabilities can have catastrophic consequences for safety. Traditional cryptographic schemes are not feasible for vehicular systems due to their diverse nature and potential insider attacks. Trust has been introduced as an alternative means of ensuring security in in-vehicle systems[5,25].

9. 1 Security Threats

Transmitting, control, and application levels face man-in-the-middle attacks, passive attacks, distributed multi-controllers, applications, unauthorized access, and security protocol conflicts. High mobility requires continuous validation components, and SDN can address potential security concerns. Addressing these challenges requires constant validation and addressing congestion in SDN-based VANETs to ensure their recognition and security. Figure 10 illustrates the possible security concerns of VANETs that can be discussed via SDN[5,58].

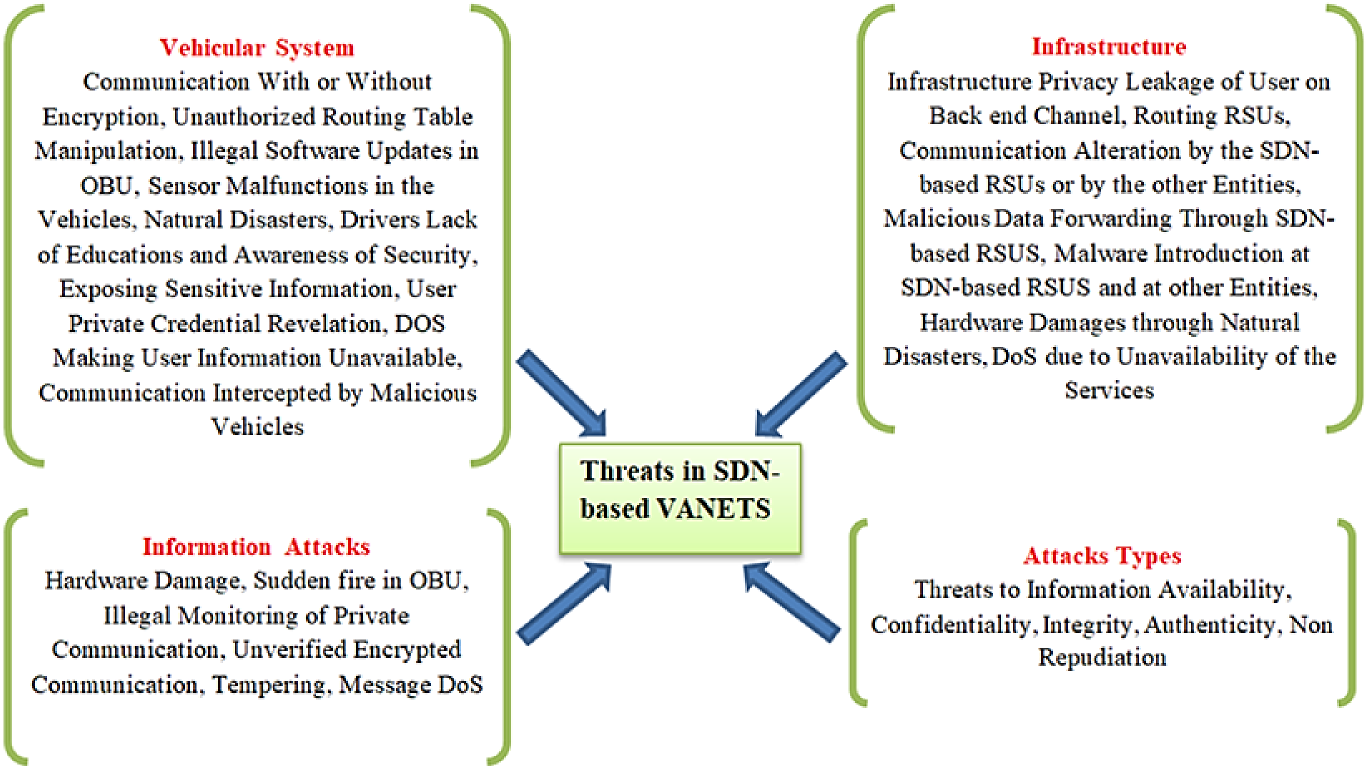
Vehicular System
Communication With or Without Encryption, Unauthorized Routing Table Manipulation, Illegal Software Updates in OBU, Sensor Malfunctions in the Vehicles, Natural Disasters, Drivers Lack of Educations and Awareness of Security, Exposing Sensitive Information, User Private Credential Revelation, DOS Making User Information Unavailable, Communication Intercepted by Malicious Vehicles
Infrastructure
Infrastructure Privacy Leakage of User on Back end Channel, Routing RSUs, Communication Alteration by the SDN-based RSUs or by the other Entities, Malicious Data Forwarding Through SDN-based RSUS, Malware Introduction at SDN-based RSUS and at other Entities, Hardware Damages through Natural Disasters, DoS due to Unavailability of the Services
Threats in SDN-based VANETS
Information Attacks
Hardware Damage, Sudden fire in OBU, Illegal Monitoring of Private Communication, Unverified Encrypted Communication, Tempering, Message DoS
Attacks Types
Threats to Information Availability, Confidentiality, Integrity, Authenticity, Non Repudiation

Figure 10: Threats taxonomy-based vehicular ad-hoc networks (VANETs) concerning vehicular systems, information, and infrastructure.

9. 2 Application-Level Threats

Toxic applications can corrupt the SDN controller and result in violation of authorization, acceleration of benefits, depletion of available resources, disruption of management chains, or injection of malicious control messages into the network, leading to catastrophic consequences in the system's behavior (such as packet dropping, rerouting, and termination of the SDN controller). Third-party apps may provide comparable actual risks because of the diversity of vendors, lack of compatibility in security solutions, and concerns over trust[5,58].

9. 3 Control Plane Threats

Interchanging switches might damage the SDN controller's perception of the network or network architecture or create artificial connections. Control messages can be manipulated to imitate system resources or get confidential information. Increasingly widespread attacks involve imposing sanctions on the SDN controller, disrupting network connectivity, or compromising controller availability. The exclusive responsibility of the controller for decision-making renders the control plane particularly vulnerable to assaults and failures. The information stored in the system can also initiate fresh assaults. Interoperability problems among controllers can also create vulnerabilities[5,58].

9. 4 Communication APIs Threats

API volatility and lack of institutionalization pose significant risks in vehicular ad hoc networks. Southbound APIs are vulnerable to man-in-the-middle attacks, eavesdropping, and unauthorized access. SDN-based VANETs require standardized APIs but lack standardization. Malicious programmable switches can expose fake system structures, while remote communication protocols and software-defined radio vulnerabilities can lead to software-layer breaches. Diverse gadgets and systems contribute to vulnerability and dissatisfaction[5,58].

9. 5 Taxonomy and Classification of Security Attacks in SDN-Based VANETs

Three main types of attackers exist: internal and external, rational and malicious, and passive and active. People who have been verified to be part of a system are considered internal attackers, while untouchable aggressors are someone else entirely. They are unbiased critics who seek to destabilize the system for their benefit. Malicious attackers intend to destroy the system without considering any benefit to themselves. While passive attackers must identify the network's existence, agile attackers can access massive amounts of data stored in any central location by creating a misleading signal. Although the communication system is not dependent on the SDN controller, it is important to consider some difficulties when building it. Table 3 summarizes the security threat categorization for SDN-based VANETs according to the three parameters mentioned before. Considering potential threats to VANETs built on SDN [5],[10,58],[20,59].

- **Hijacking of Session.** The authentication procedure is initiated at the commencement of the session. Once the connection is established, session hijacking becomes straightforward. In this type of assault, the perpetrators get comprehensive knowledge regarding the session and subsequently position themselves as the center node among the nodes[6],[10,20].

- **Disclosure of Identity.** The driver is the vehicle's owner and may utilize personal information throughout the verification procedure. Consequently, it is simple for intruders to infiltrate the system[10].
- **Geolocation Monitoring.** The vehicle's position might be utilized to monitor it and acquire information regarding the driver and occupants[10].
- **Eavesdropping.** Surreptitiously monitoring conversations without consent[10],[59,18].
- **Denial of Service.** This is the most significant assault. The assailants obstruct a particular node from utilizing services[10],[59,18].
- **Jamming Attack.** This approach involves the attacker obtaining information regarding the frequency of signals received by the receiver and subsequently transmitting a signal at the same frequency to obstruct the genuine signals[10,59],[20,18].
- **Distributed DOS attack.** In this form of assault, numerous attackers target a specific node to obstruct its access to services[6],[10,59].
- **Repudiation Attack.** Attempting to refute the transmission or reception of a message[59],[18,19].
- **Masquerading Attack**. Attackers often take on the persona of a different entity[59,18].
- **Illusion Attack**. Creating false signals or scenarios to deceive people[59],[18,23].
- **Sink-Hole Attack.** Compromising a node to divert network traffic[59],[18,23].
- **Replay Attack**. Replicating legitimate data streams to obtain unauthorized access[59],[18,23].
- **Free Riding Attack**. Exploiting resources without providing input or according to regulations[59,19].
- **Message Tampering Attack**. Modifying the messages that are being conveyed[59,18].
- **Impersonation Attack**. Impersonating a legitimate user[59],[18,23].
- **Sybil Attack.** Generating several fictitious identities to influence a network[59],[20,18].
- **Gray Hole Attack**. Partially discarding packets to impede communication[20,59].
- **Greedy Behavior Attacks.** Exploiting the system for individual advantage[59,19].
- **Broadcast Tampering Attack.** Modifying or disrupting broadcast communications[19].
- **Man-in-the-middle attack.** Intercepting communication between vehicles or between vehicles and infrastructure[59],[18,23].
- **Traffic Analysis Attack.** Analyzing data flow patterns to obtain intelligence[18,19].
- **Timing Attacks.** Exploiting the time required to perform cryptographic algorithms[5,18].
- **GPS Spoofing Attack.** Manipulating geographic data by emitting erroneous GPS signals[59],[18,19].
- **Black Hole Attack**. Receiving network traffic without sending it[20],[59,19].
- **Malware Attack.** This attack employs software components to manipulate and gain control over the OBUs and RSUs. As a result of this attack, the components of VANETs begin to experience malfunctions[18,19].
- **Spamming Attack**. Spamming the system with unwanted communications[5,19].
- **Rule Conflicts Attack**. A Rule conflict attack has the potential to compromise confidentiality by enabling unauthorized individuals to gain access to or reveal sensitive information through the exploitation of conflicting or inadequately implemented security rules[5].
- **Host Location Hijacking**. This attack involves the attacker impersonating the target host to intercept their traffic. An attacker compromises the controller's HTS and alters the target host profile database's parameters. Attacks like this effect topology-dependent applications like load balancing and routing. It is also capable of forging LLDP packets and sending them to the controller as fake link records [60,61].
- **Switch-based attack.** Through malicious OpenFlow switches, an attacker can launch a topology-poisoning assault. These infected switches send bogus LLDP signals showing fake connections in the network. The lack of evidence on creating fake links in SDN makes it challenging to detect this attack, which is known as a link fabrication attack [60,61].

fake LLDP packet injection. LLDP packet contents are altered and transmitted among RSUs within its SDN-based vehicular architecture[60]**.**

Table 3: SDN-based VANET Attack Classification

Type of Attack	Attackers from both inside and outside	Categorize based on Rational or Malignant attack	Impact of the Attack
Hijacking of Session	Outside /Inside Attack	Malignant attack	Active
Disclosure of Identity	Inside Attack	Rational attack	Passive
Geolocation Monitoring	Outside Attack	Malignant attack	Passive
Eavesdropping	Outside Attack	Malignant attack	Passive
Denial of Service	Inside Attack	Malignant attack	Active
Jamming Attack	Inside Attack	Malignant attack	Active
Distributed DOS attack	Inside Attack	Malignant attack	Active
Repudiation Attack	Inside Attack	Rational attack	Passive
Masquerading Attack	Inside Attack	Rational attack	Active
Illusion Attack	Inside Attack	Malignant attack	Active
Sink-Hole Attack	Outside /Inside Attack	Malignant attack	Active
Replay Attack	Outside /Inside Attack	Malignant attack	Active
Free Riding Attack	Inside Attack	Rational attack	Passive
Message Tampering Attack	Inside Attack	Malignant attack	Active
Impersonation Attack	Outside / Inside Attack	Rational attack	Active
Sybil Attack	Inside Attack	Malignant attack	Active
Gray Hole Attack	Inside Attack	Malignant attack	Active
Greedy Behavior Attacks	Inside Attack	Rational attack	Active
Broadcast Tampering Attack	Inside Attack	Malignant attack	Active
Man-in-the-Middle Attack	Outside /Inside Attack	Malignant attack	Passive
Traffic Analysis Attack	Outside Attack	Rational attack	Passive
Timing Attacks	Outside Attack	Malignant attack	Active
GPS Spoofing Attack	Outside Attack	Malignant attack	Active
Black Hole Attack	Outside Attack	Malignant attack	Active
Malware Attack	Outside Attack	Malignant attack	Active
Spamming Attack	Outside Attack	Malignant attack	Active
Rule Conflicts Attack	Inside Attack	Malignant attack	Active
Host Location Hijacking	Outside /Inside Attack	Malignant attack	Active
Switch-based attack	Outside /Inside Attack	Malignant attack	Active
fake LLDP packet injection	Outside /Inside Attack	Malignant attack	Active

9. 6 CIA3 Security Model for SDN-based VANET Attacks

This section identifies critical vulnerabilities and classifies them based on the CIA3 model. This approach provides a deeper understanding of how security threats affect the availability, confidentiality, and authentication across different planes. Through several groundbreaking ideas, SDN technology can significantly enhance contemporary networking. Nevertheless, numerous security domains, challenges, and opportunities have emerged in this new era based on SDN. The following sections of the CIA3 exam deal with SDN security.

9. 6. 1 Ensuring Confidentiality in SDVN Communication

Confidentiality is essential in communication, particularly in wireless networks, where man-in-the-middle (MitM) assaults may be utilized. Advanced security measures are crucial in modern network technologies such as Software-Defined Networking (SDN), which help alleviate these threats. The SDN-based security infrastructure preserves connections on dedicated lines to ensure confidentiality, while centralized control of encryption and key management methods is implemented across the network. The absence of intrinsic security in north-bound

API protocols renders SDN susceptible to sniffer attacks, necessitating the implementation of Secure Socket Layer (SSL) encryption and public key certificate infrastructure. The lack of standardized or open specifications for Northbound APIs and multi-vendor strategies presents considerable dangers to user data and network confidentiality in SDN. Contextual security elements are provided to augment network security and mitigate Man-in-the-Middle attacks [66-69].

9. 6. 2 Integrity Assurance Across Control and Data Planes

Data integrity is essential in all communication contexts. SDN has established several unique methods to ensure data integrity in VANET connections. SDN-based contextual data integrity paradigm in which the SDN application collects contextual information from storage partitions to validate the necessary context. Data modification and data flow modification occur in the control and data layers. API exploitation occurs when hackers exploit the communication mechanisms within the control plane, including between controllers and other devices. A bot management system is essential to combat this, ensuring only legitimate users can access APIs. This solution should include comprehensive threat identification, thorough reporting, and adaptable implementation options[64],[70-72]. Consequently, meticulous oversight of network data records is essential for maintaining the integrity of the network in an SDN environment. Similarly, the south-bound API protocol, OpenFlow, is prone to deterioration in data integrity, particularly on wireless networks[2].

9. 6. 3 Authentication Mechanisms for Identity Validation in SDVNs

The modern communication framework has substantial privacy challenges from unapproved smart gadgets and rudimentary text-based verification techniques. SDN, a centralized and high-capacity computational architecture, empowers modern networks to meet sophisticated authentication systems' computational and processing requirements efficiently. SDN can mitigate computational challenges in resource-constrained nodes facilitated by MEC. The SDN-based design alleviates communication overhead issues in essential smart grid system applications. A worldwide viewpoint on implementing hierarchical authentication measures across vast networks in SDN. The software-based architecture of SDN may integrate all traditional cryptography techniques, including hash tables and hash functions. A cohesive, diverse, backward-compatible, and universal authentication system must be devised for the SDN-based network. Considering the importance of the SDN controller, it is essential to implement a strong authentication mechanism. SDN serves as a feasible alternative for security and authentication in wireless body area networks because of its distinctive robustness, centralized control, and flexibility [73-77].

9. 6. 4 Availability Against DoS Attacks in Software-Defined VANETs

Availability depends on the network's capacity to withstand all DoS attacks. A comprehensive overview of the deployment of Software-Defined Networking (SDN) provisions, incorporating diverse, innovative DDoS detection methodologies alongside conventional and unconventional technologies, including information theory, machine learning, and artificial intelligence[2].In [78], it investigates the application of SDN for the detection of Distributed Denial of Service (DDoS) attacks, utilizing machine learning (ML) and deep learning (DL) methodologies to deploy adaptive security protocols that react to threats instantaneously. Through constant surveillance of network traffic and modifying security protocols, SDN can mitigate service interruptions resulting from DDoS attacks, improving the overall availability of IoT services[79].[80] present a practical and lightweight detection and mitigation technique tailored for the SDN controller in SD-IoV setups. The efficacy of the proposed approach is assessed based on detection time, accuracy, mitigation efficiency, controller load, and connection bandwidth utilization. The [81]examines the detection and mitigation of DDoS assaults within the SDN framework, utilizing Open vSwitch specifications and Python for traffic analysis. Thus, it demonstrates the practical implementation of SDN in real-time contexts. These attacks can severely affect the availability of network services by saturating resources and obstructing genuine users from accessing the network[82-85].

9. 6. 5 Adaptive Access Control Strategies in SDN-Based Vehicular Networks

The SDN-based architecture routinely enforces access control techniques. Regarding mobile and IoT applications, SDN is a good fit for static and dynamic access control. To set up a thorough centralized network access control, SDN allows dynamically modifying flow rules in network switches. But there are significant security concerns with the north-bound API and related interfaces due to the centralized architecture of SDN. An Intrusion Detection System (IDS) based on anomalies will be used to find suspicious activity in the network. This innovation allows for the creation of software-defined flow rules that may adjust to the current state of the network in real time, making access control systems more adaptive and responsive. To guarantee the exact implementation of the intended network behavior, policy analysis, verification, and improvement are of utmost importance. Administrators can better manage access control for network security with the help of this methodology [88].

Provide an efficient, lightweight intrusion detection system that employs blockchain technology to mitigate attacks in SDN/NFV-enabled cloud environments. The proposed system aims to enhance detection rate, accuracy, precision, recall, and authentication time, enabling a secure environment for 5G users within the SDN/NFV-enabled cloud. BENBI is a scalable and dynamic access control system designed for the Northbound Interface of SDN-based VANETs. It addresses security issues, including resource exposure and configuration alterations, while maintaining the confidentiality and integrity of transmitted communications [88]. The [90] This paper introduces a mechanism for the dynamic configuration of access control lists to enhance security in Software-Defined Networking environments. This approach mitigates detrimental traffic, particularly during DoS assaults, by allowing the SDN controller to dynamically manage traffic flow based on real-time conditions.

9. 7 Control Plane Security Solutions in SDN-Enabled VANETs

The SDN paradigm enables automation and flexibility, essential attributes required to address challenges now faced in network operations. Consequently, SDN separates the control plane, where control programs determine the routing of network packets, from the data plane, where the forwarding hardware executes these decisions. Before the advent of SDN, control programs were intricately linked with each device dispersed over the network [93]. Tatang et al., 2017 present SDN-GUARD as a novel system designed to detect and mitigate SDN rootkits that compromise SDN controllers. The author examines security options in SDN. It classifies methodologies into conventional security measures, AI-driven procedures, and Moving Target Defense (MTD) strategies. Abdi et al. 2024 emphasize the risks intrinsic to the centralized architecture of SDN and suggest the SAFETY Solution (SDN-Assisted Firewall for Enforcing Topology Constraints. Security focuses on topology-based access control and firewall implementation [75].

The document presents SLICOTS, a countermeasure against TCP SYN flooding attacks in SDN. It leverages the programmability of SDN to monitor and obstruct malicious TCP requests, hence decreasing response time and augmenting SDN security. SLICOTS (Secure and Lightweight Countermeasure for TCP SYN Flooding Attacks) was selected for its efficacy in detecting and mitigating SYN flooding at the SDN control plane with minimal processing cost. Its lightweight architecture renders it particularly appropriate for resource-limited VANET applications where reducing latency and controller burden is essential [94]. Another author provides the Flood Guard framework to counter data-to-control plane saturation assaults in SDN settings. Flood Guard was chosen to directly mitigate data-to-control plane saturation attacks, a prevalent and detrimental danger in SDN systems. Its proactive rule-analysis methodology and packet relocation capacity sustain availability and avert controller resource depletion, essential for the reliability of time-sensitive vehicular applications. The framework utilizes proactive flow rule analysis and packet relocation to alleviate the effects of these attacks. The system dynamically produces flow rules and stores flooding packets, guaranteeing network operation while reducing resource usage [95].

Other authors in [96] propose alleviating control plane saturation threats in Software-Defined Networking (SDN). Present LineSwitch, which utilizes probabilistic proxying and blacklisting to safeguard the control plane while preserving network functionality. Line Switch provides an innovative approach to control plane dependability through probabilistic proxying and switch-based blacklisting. It was chosen for its efficacy in alleviating buffer saturation and control-message flooding, while maintaining performance. The concept is well-suited for SDN-enabled VANET applications, where it is crucial to maintain flow consistency amid varying topologies. SDNShield uses network function virtualization (NFV) to establish a scalable and adaptive DDoS defense mechanism. It is selected for its three-tier defense architecture, which offers adaptive protection, TCP handshake verification, and flow-based traffic analysis. This renders it highly appropriate for dynamic and high-mobility environments such as SDVNs, where risks necessitate real-time mitigation [51]. The [97] introduces SGS, a Safeguard Scheme aimed at safeguarding the control layer of SDN against DDoS attacks. It presents a dual-module framework: anomaly traffic detection and dynamic defense. The initial module detects fraudulent traffic via a four-tuple feature vector, while the subsequent module reallocates controllers to alleviate the effects of attacks. SGS markedly improves detection precision and diminishes flow configuration and controller response durations.

Ruffy et al. in [98] examine essential security vulnerabilities and suggest recommendations for developers to improve security in SDN implementations. The STRIDE threat model is employed to discover vulnerabilities in SDN components, highlighting the necessity for a secure design. Sahay et al.in [99] introduce ArOMA, a Software-Defined Networking framework for autonomous Distributed Denial of Service mitigation. ArOMA unifies multiple security features by utilizing the programmability of SDN. It effectively sustains video streaming performance amid DDoS attacks, illustrating its potential for scalable and automated defense measures. ArOMA signifies a substantial progression in network security. also, Tchendji et al. present in [100] E2BaSeP is a Bayes-Based Security Protocol, a novel security mechanism devised to counter ARP spoofing attacks in Software-Defined Networking (SDN). It utilizes a Bayes-based detection algorithm to ensure efficient attacker identification while preserving a secure Global ARP Cache. It can improve network security in both static and dynamic contexts.

The Moving Target Defense (MTD) system augments Software-Defined Networking (SDN) security by constantly modifying network configurations, complicating the exploitation of vulnerabilities by attackers. This method encompasses approaches like randomizing IP and MAC addresses, port numbers, and flow tables [75]. Ahmed et al. present in [101] the U-TRI Technique (Unsupervised Learning-based Threat Response and Identification) to augment network security, especially within SDN contexts. It utilizes unsupervised learning to detect and address unidentified risks without dependence on pre-labeled data. This methodology seeks to enhance the precision of threat identification and diminish false positives, hence overcoming the shortcomings of conventional intrusion detection systems. the comparative table consolidates recent research on SDN control plane security, highlighting advantages such as AI-driven detection and dynamic reaction while recognizing challenges related to deployment complexity and scalability constraints. Table 4 examines the security emphasis, protective methods, benefits, and drawbacks of several SDN control plane security solutions.

Table 4: Comparative Table of SDN Control Plane Security Solutions.

Solution	Security Focus	Protection Mechanism	Advantages	Limitations
SDN-GUARD	Anomaly detection, authentication	Machine learning-based threat detection	AI-powered detection, real-time response	High processing overhead, potential false positives
Safety	Topology-based security, access control	SDN-assisted firewall enforcement	Strong access control prevents rule injections	Requires detailed security policies
SLICOTS	Lightweight control plane security	Encrypted control traffic, authentication	Low overhead, efficient security mechanisms	It may not scale for large SDN deployments
FloodGuard	DDoS mitigation	Traffic anomaly detection, rate limiting	Protects controllers from overload	It may not stop sophisticated botnet-based DDoS
LineSwitch	Control plane reliability	Switch-based security policies	Enhances fault tolerance, prevents routing attacks	It may require SDN switch modifications
SDNShield	Controller Security	API monitoring, authentication mechanisms	Prevents unauthorized access and modifications	It may introduce latency due to monitoring
Safe-GUard	Real-time monitoring	Intrusion detection, automated response	Instant attack mitigation	Requires advanced monitoring capabilities
PackedChecker	Flow rule verification	Packet verification before rule application	Prevents malicious rule injections	Limited scalability for high-flow networks
STRIDE	Threat classification	Categorization of attack vectors	Comprehensive risk assessment	It does not provide direct mitigation
ArOMA	Multi-layer security	Adaptive security policies	Dynamic response to threats	High complexity in deployment
Bayes-Based Protocol	Anomaly detection	Bayesian probability analysis	Detects zero-day attacks	Requires large datasets for accuracy
Moving Target Defense	Dynamic attack prevention	Randomized configurations	Unpredictable attack surface	High processing and operational overhead
U-TRI Technique	AI-driven threat response	Unsupervised learning	Detects new attack patterns	Computationally expensive
ASLB Architecture	Secure load balancing	Adaptive load balancing	Prevents controller bottlenecks	May introduce traffic delays
FS-Open Security	Multi-layer security framework	Policy-driven security mechanisms	Customizable security policies	Requires manual policy tuning
Controller-to-Controller Protocol	Secure inter-controller communication	Encrypted messaging between controllers	Ensures redundancy and failover support	Limited to multi-controller environments
McNettle	High-performance controller security	Real-time SDN traffic analysis	Low-latency security enforcement	Limited to high-speed SDN environments
HyperFlow	Controller redundancy	Synchronization between multiple controllers	Prevents single points of failure	Requires additional network resources

9. 8 Security Evaluation Metrics in SDVNs

To evaluate the efficiency of security solutions in SDN-based VANETs, researchers usually employ the following metrics [51], [94-97]:

- **Detection Accuracy:** Measures the ability of the system to correctly identify malicious actions or anomalies.
- **False Positive Rate (FPR):** The rate at which routine conduct is wrongly classified as an attack.
- **Response Time / Latency:** In automotive scenarios, the system's time to notice and respond to an attack is crucial.
- **Controller Load / Resource Utilization:** assesses the processing overhead a security solution adds to the SDN controller.
- **Packet Loss Rate:** Used to analyze DoS/DDoS impact and countermeasure performance.
- **Throughput / Bandwidth Usage:** Measures how assaults or defense strategies affect normal data flow.
- **Flow Table Utilization:** Indicates vulnerability to saturation attacks, such as table overflows in OpenFlow switches.

For instance, technologies like Flood Guard and LineSwitch have exhibited excellent detection accuracy (up to 97%) while keeping low controller overhead. Similarly, SLICOTS efficiently mitigates SYN flooding with low latency impact, and SDNShield accomplishes a fair trade-off between detection precision and controller reaction time, utilizing NFV-based traffic filtering.

10 Benefits of the SDN-based VANET Implementations

The Software Defined Network significantly streamlines network management by efficiently utilizing resources through global network information. While SDN offers several advantages, the fundamental attributes of SDN architecture, such as programmable SDN-based switches, the restricted bandwidth of the southbound channel, and constrained resources at SDN controllers, can give rise to new security risks [6].The SDN-based VANET system has notable benefits compared to conventional VANET systems, including handling network transmission without human configuration. It offers a comprehensive global system perspective, providing enhanced security procedures. In contrast to traditional VANET systems that involve extensive information sharing, the centralized controller gathers traffic flow data[10]. The following are the most significant advantages of integrating SDN-based VANET.

- **Network-wide intrusion detection.** An intrusion detection system (IDS) scrutinizes the traffic patterns gathered by the router or cars to identify malicious activity. The SDN controller formulates policies and regulations for the RSU, which are subsequently transmitted to all linked cars[10]. These two categories of detection methodologies are used in intrusion detection systems (IDSs).

i. Intrusion Detection Systems Based on Signatures: These systems store a database of known or predicted attack patterns or signatures, which they compare to the packets' signatures to determine if they are malicious [30].

ii. Intrusion Detection Systems Based on Anomalies: These systems seek out malicious software that has not yet been identified. They pick up on things that don't fit the expected pattern. To identify potentially harmful or suspicious incoming network traffic, the IDS uses a trained machine-learning model [30].

- **Detection of Malicious Behavior by RSU or Vehicle.** The SDN controller facilitates the identification of malevolent data traffic and actions conducted by RSU or vehicle nodes. Conventional networks employ routing protocols to accomplish this objective, but SDN-based systems simplify the process by transmitting comprehensive data on received, forwarded, and discarded packets. The controller efficiently identifies malfunctions in RSUs during their communication, enabling effective detection[10,91].

- **Network Forensics.** A subfield of digital forensics is employed to oversee data transmission inside a network. Data or traffic flow information is collected to identify any errors. It is utilized to monitor the flow of network traffic. If an assailant deletes or modifies the settings of a device, network-based evidence is employed during the forensic examination[10].

- **Self-Healing Mechanisms.** SDN-based VANET stands out from basic VANET systems due to its built-in self-healing capability. The controller sets rules for all devices and nodes, which nodes can meet to prevent attacks. Self-healing involves automatic recovery after identifying harmful behavior and restoring the system to its original state in case of an attack[10].

- **Path Selection.** The SDN-based technology simplifies the process of determining the most efficient routing. In VANETs, data traffic distribution might become uneven due to using the shortest routing channel. When video streaming occurs, the node occasionally utilizes a substantial bandwidth. If this scenario arises inside the SDN-based VANET system, the SDN controller will redirect the traffic flow to optimize the network's efficiency and minimize the likelihood of congestion[10,91].

- **Selection of Channel and Frequency.** Data transmission can be carried out via wireless channels operating at various frequencies. The SDN system enables the SDN controller to choose a suitable channel for transmitting data. The SDN controller dynamically chooses the optimal frequency for data transmission at any given moment. Emergency communications are transmitted using a designated frequency channel[10,91].

- **Support for heterogeneity and improved resource utilization.** SDN architecture enables device heterogeneity by utilizing its standard programmable interface, such as Open Flow. This means that networking devices from various vendors can interact with each other and control plane entities as long as they are configured with open communication interfaces, such as the OF protocol. SDN can improve network resource management by making intelligent decisions, such as adjusting transmit power for successful packet delivery. However, in a VANET environment, SDN faces challenges in efficiently managing resources across multiple RSUs, and frequent switching issues can negatively impact performance[6,12],[91,92].

- **Improved network security.** The controller can get crucial network information by talking with the OF-Switches. These switches can gather the necessary data through network traffic analysis and utilize diverse anomaly-detection techniques. Subsequently, the controller examines and establishes connections between the reactions of the data plane entities to generate or modify its comprehensive network perspective. Upon analyzing the results, it is possible to implement new settings and rules throughout the network to prevent the observed or anticipated security concerns. Therefore, implementing these methods might enhance the efficiency of the network and expedite the management and prevention of detected security weaknesses[6],[12,92].

- **Single point of failure.** The distributed denial of service (DDoS) attacks can target SDN in many ways because of its centralized controller, the small size of its flow tables, and the limited bandwidth of the communication path between the controller and the OF-Switch. Furthermore, there are still no established best practices for software-defined networking functions and components, and the lack of trust among data plane entities is a direct outcome of networks' support for open programmability[6,12].

- **Minimizing service latency.** optimizes fog computing on network edge routers, reducing latency for time-sensitive applications. Its programming flexibility allows for seamless implementation, with proposals including genetic algorithms, centralized routing, mobility prediction, artificial intelligence-powered controllers, and dynamic reconfiguration of router flow tables for adaptive networking services[6,12].

- **Network slicing.** The network slicing was offered as a technique for efficiently managing network resources. It involves the creation of virtual network layers, referred to as slices, that aggregate services and applications with shared characteristics. Each slice functions as an autonomous network, meaning slices are segregated from one another and administered independently, with allocated network resources. The slicing enhances the visibility of network resources and user needs. Consequently, the SDN control will be enhanced by establishing suitable regulations for each slice according to its characteristics[13].

- **Fast and flexible network configuration.** Delineating control and logic planes in Software-Defined Virtual Networks facilitates swift and adaptable network deployments. It will assist in addressing the diverse demands of applications and accommodate alterations in network topology resulting from vehicle mobility. A data-driven methodology for developing an artificial intelligence model for predicting vehicular traffic behavior. Specifically, they integrate the flexibility, adaptability, and scalability of SDVN infrastructures with machine learning approaches to simulate traffic flow effectively[12].

10. 1 Utility of SDN-Based VANETs Applications

To ensure that passengers have a safe travel experience, cars can communicate with one another and share various information. The driver can make a more informed judgment in an accident. Through integration, several types of sensors enable the car to gather and analyze data, which is then utilized to enhance the vehicle's and its occupants' safety. This classification of safety and efficiency services demonstrates the real-world relevance of SDVN solutions, bridging theoretical models with practical vehicular applications such as emergency response, smart parking, and collision avoidance systems. Table 5 shows the classification of VANET application services into two groups: safety and efficiency[5,10]. The integration of SDN is advantageous in several vehicular network scenarios[13]. Efficient data distribution: A prominent characteristic of SDVNs is their highly customized data dissemination. Data dissemination enhances several services, including emergency broadcast services, adaptive broadcast interval timing, and user security[8].The implementation of SDN prioritizes emergency services. Called SPArTaCus, prioritizes emergency services in smart cities by implementing a priority management layer within the SDN architecture [102].The introduction of a hybrid emergency message delivery strategy in the Internet of Vehicles (IoV) by utilizing SDN. The goal is to ensure the swift and dependable transmission of emergency communications[102]. The adaptability, real-time capabilities, and high responsiveness are among the reasons for its broad range of applications[8]. One of the applications is Intersection Collision Avoidance. This system utilizes V2I communication to assist the vehicle or driver decide when to cross an intersection. A Remote Unit (RU) collects data from nearby moving cars and analyzes it for potential warnings or accidents, as illustrated in Figure 11 [5].

Table 5: Application and Services of SDN-Based VANET System.

Application	Type	Latency Sensitivity	Required SDVN Component	Communication Mode
Comfort/Entertainment Applications	Non-Safety	Low	Controller, RSU	V2I / V2C
Intersection Collision Avoidance	Safety-Critical	High	Controller, RSU	V2V
Warning in Case of Violating Stop Signal	Safety-Critical	High	Controller, RSU	V2V / V2I
Stop Movement Sign Assistant	Safety-Critical	High	RSU, SDN Switch	V2I
Sign Extension	Safety-Enhancement	Medium	RSU, Controller	V2I
Warning in Case of Blind Merge	Safety-Critical	High	RSU, Controller	V2V / V2I
Pedestrian Intersection Information and Crossing Road Warning	Safety-Critical	High	RSU, SDN Virtual Switch	V2P / V2I
Emergency Vehicle Is Approaching Warning	Safety-Critical	High	RSU, Controller	V2V / V2I
Emergency Vehicle Signal Pre-emption	Safety-Critical	High	RSU, Bandwidth Manager	V2I
Post-Crash Warning	Safety-Critical	High	RSU, Controller	V2V

Cooperative Forward Collision Warning	Safety-Critical	High	Controller, OBU	V2V
Road Condition Warning	Safety-Enhancement	Medium	RSU, Controller	V2V / V2I
Left/Right Turn Assistant	Safety-Critical	High	RSU, OBU	V2V
Lane Change Warning	Safety-Critical	High	Controller, OBU	V2V
Traffic Accident Detection	Safety-Critical	High	RSU, Virtualized Network Functions	V2I / V2C
Remote Video Analysis and Initial Assessment	Non-Safety	Medium / High	RSU, Bandwidth Manager	V2I / V2C
Service Priority Adaptiveness for Emergency Traffic Using SDN (SPArTaCuS)	Safety-Critical	High	Central SDN Controller	V2I / V2C
SDN-Enabled Hybrid Emergency Message Transmission Architecture in IoV (HEMT)	Safety-Critical	High	RSU, Hybrid SDN Controller	V2V / V2I
Next Generation Emergency Communication Systems via SDN (NGECS)	Safety-Critical	High	Next-Gen SDN Controller, RSU	V2I / V2C
Latency-based Routing	Efficiency Services	High	SDN Controller, RSU	V2I / V2C
Smart Grid Application	Efficiency Services	Medium / High	RSU, SDN Controller	V2I
Heterogeneous Support	Efficiency Services	Medium	RSU, Multi-Access Gateway	V2I / V2V
Fast and flexible network configuration	Efficiency Services	High	SDN Controller, Virtual Switch	V2V / V2I
Handover in VANETs	Efficiency Services	High	RSU, Mobility-Aware Controller	V2V
Efficient data dissemination	Efficiency Services	High	RSU, SDN Broadcast Modules	V2V / V2I
Virtualization of wireless network	Efficiency Services	Medium / High	Virtual Network Functions (VNFs), Controller	V2I / V2C
Smart parking	Efficiency Services	Medium	RSU, Parking Sensors, Cloud Controller	V2I / V2C
Bandwidth management	Efficiency Services	Medium / High	SDN Controller, Bandwidth Allocator	V2I / V2C

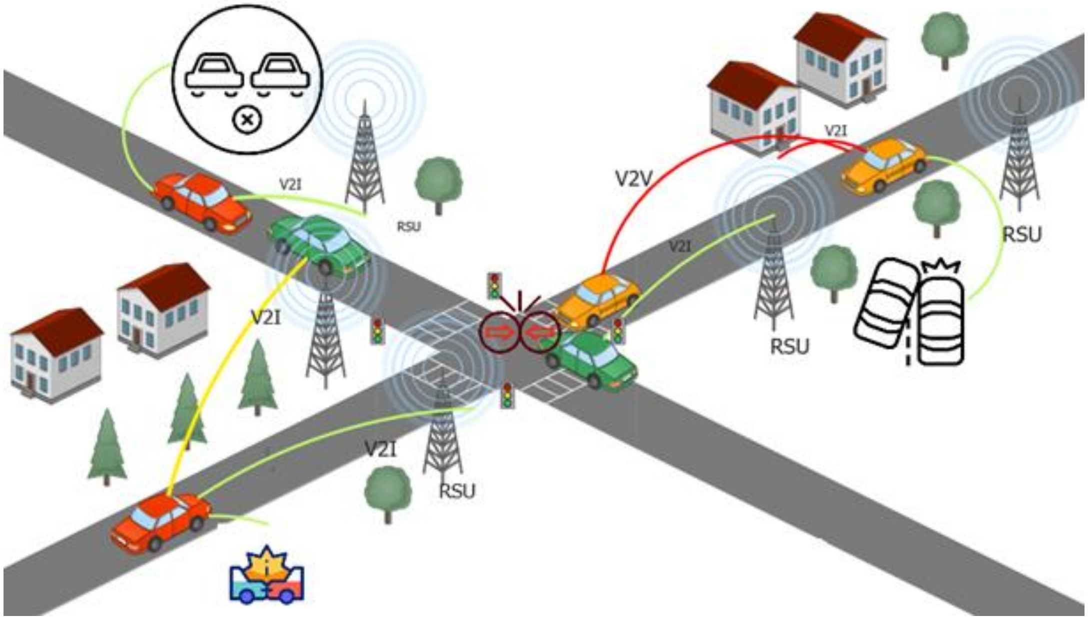

Figure 11: Intersection Collision Avoidance services in SDVN.

Proper deployment of the SDVN system can initiate an industrial revolution in its domain. Vehicular networks predominantly drive Intelligent Transportation Systems (ITS), which comprise numerous succinct applications capable of transforming the automotive sector. Figure 12 illustrates several applications of SDVN that have been previously deployed.

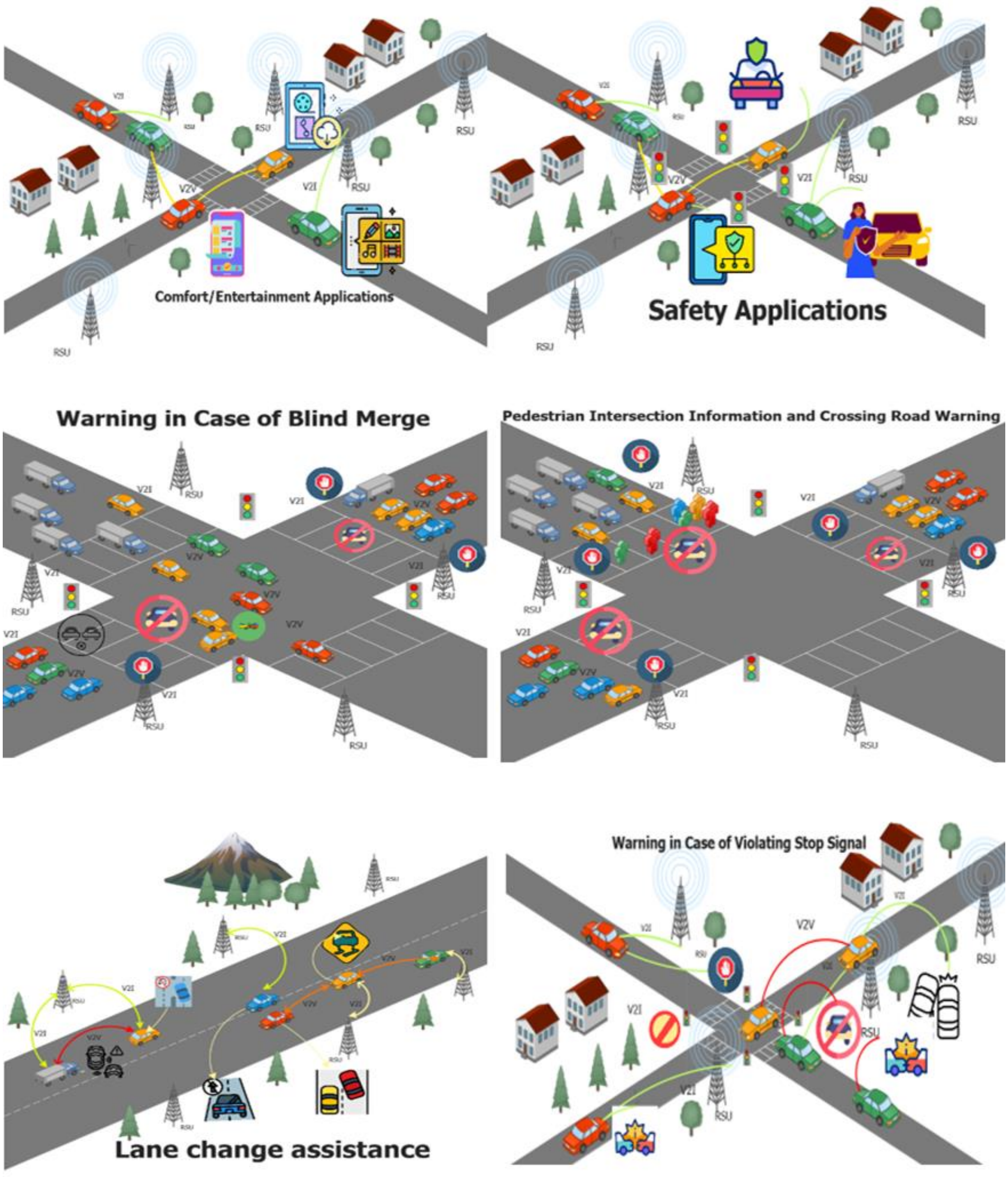
Comfort/Entertainment Applications
Safety Applications
Warning in Case of Blind Merge
Pedestrian Intersection Information and Crossing Road Warning
Lane change assistance
Warning in Case of Violating Stop Signal
RSU
V2I
V2V

Figure 12: Different services of software-defined vehicle networks (SDVN).

11 Conclusion and future work

This survey provides a comprehensive and organized examination of the security framework of Software-Defined VANETs (SDVNs), focusing on practical implementation and security safeguarding techniques. The study classifies over 30 assaults through a multidimensional taxonomy and aligns them with the SDVN architecture, providing valuable insights for the development of resilient and secure vehicular communication systems. The comparative assessment of countermeasures such as SDNShield, SLICOTS, Flood Guard, and LineSwitch offers actionable alternatives for engineers and developers to enhance network resilience against significant threats, including DoS and routing manipulation. The use of the CIA3 model enables practitioners to identify which elements of network security—confidentiality, integrity, authentication, availability, or access control—are vulnerable in specific scenarios, thereby facilitating risk-based security planning. Moreover, the survey's results are pertinent to essential vehicular applications, including emergency communication, collision avoidance, intelligent traffic routing, and secure autonomous driving systems.

This work functions as a reference framework for system architects, security analysts, and policymakers engaged in the development of next-generation Intelligent Transportation Systems (ITS). Subsequent research should build upon this groundwork by developing lightweight, real-time, and AI-driven security mechanisms capable of scaling within increasingly complex and autonomous automotive ecosystems. The paper further evaluates contemporary security countermeasures, including SDN-GUARD, SDNShield, Flood Guard, and Moving Target Defense strategies, providing a comparative analysis based on deployment feasibility, detection capabilities, and overhead. The CIA3 model assesses these solutions across key security pillars: Confidentiality, Integrity, Authentication, Availability, and Access Control. Moreover, the practical implications of SDVNs are explored through a review of service-oriented applications in safety, efficiency, and infotainment, emphasizing their importance in real-time traffic management and emergency response systems.

While SDN offers a centralized and programmable approach to network control, its deployment in vehicular environments introduces new challenges related to controller bottlenecks, communication overhead, and susceptibility to targeted attacks. To address these, future research should prioritize the development of lightweight cryptographic protocols, federated trust models, and AI-powered intrusion detection systems tailored for highly dynamic vehicular settings. Blockchain integration for decentralized identity management and adopting standardized API protocols across controllers also represent promising directions. By addressing these challenges, researchers can pave the way for more robust, secure, and efficient vehicular network infrastructures.

Table 6: List of abbreviations.

Abbreviations	Description	Abbreviations	Description
VANET	Vehicular Ad Hoc Networks	V2V	vehicle-to-vehicle
ITS	Intelligent transportation systems	V2I	vehicle-to-infrastructure
RSUs	Roadside Units	V2X	Vehicle–to–everything
OBU	On-Board Unit	SDVNs	Software-defined vehicular networks
DSRC	Dedicated Short-Range Communication	MTD	Moving Target Defense
SDN	Software-Defined Network	SDMN	Software-Defined Mobile Networks
API	Application Development Interface	CAVs	Connected and autonomous vehicles

References

[1] M. K. Murtadha and B. M. Mushgil, "Flexible handover solution for vehicular ad-hoc networks based on software defined networking and fog computing," *Int. J. Electr. Comput. Eng.* **13**(2), 1570–1579, doi: 10.11591/ijece.v13i2.pp1570-1579 (2023).

[2] S. H. A. Kazmi, F. Qamar, R. Hassan, K. Nisar, and B. S. Chowdhry, "Survey on Joint Paradigm of 5G and SDN Emerging Mobile Technologies: Architecture, Security, Challenges and Research Directions," *Wirel. Pers. Commun.* **130**(4), 2753–2800, doi: 10.1007/s11277-023-10402-7 (2023).

[3] N. Aljeri and A. Boukerche, "Mobility Management in 5G-enabled Vehicular Networks," *ACM Comput. Surv.* **53**(5) doi: 10.1145/3403953 (2020).

[4] R. Duo, C. Wu, T. Yoshinaga, J. Zhang, and Y. Ji, "SDN-based handover scheme in cellular/IEEE 802.11p hybrid vehicular networks," *Sensors (Switzerland)* **20**(4), 1–17, doi: 10.3390/s20041082 (2020).

[5] M. Arif *et al.*, "applied sciences and Challenges," *Appl. Sci.***10**(9), 2020.

[6] W. Ben Jaballah, M. Conti, and C. Lal, "A Survey on Software-Defined VANETs: Benefits, Challenges, and Future Directions," 2019, [Online]. Available: http://arxiv.org/abs/1904.04577

[7] L. Nkenyereye, L. Nkenyereye, S. M. Riazul Islam, Y. H. Choi, M. Bilal, and J. W. Jang, "Software-defined network-based vehicular networks: A position paper on their modeling and implementation," *Sensors (Switzerland)* **19**(17), 1–14, doi: 10.3390/s19173788 (2019).

[8] J. Bhatia, Y. Modi, S. Tanwar, and M. Bhavsar, "Software defined vehicular networks: A comprehensive review," *Int. J. Commun. Syst.* **32**(12), 1–22, doi: 10.1002/dac.4005 (2019).

[9] A. Anjum and A. A. Khan, "Optimized Communication in 5G-Driven Vehicular Ad-hoc Networks (VANETs)," 2019.

[10] H. Shafiq, R. A. Rehman, and B. Kim, "Services and Security Threats in SDN Based VANETs: A Survey," **2018**, 2018.

[11] F. H. Rahman, S. H. Shah Newaz, T. W. Au, W. S. Suhaili, and G. M. Lee, "Off-Street Vehicular Fog for Catering Applications in

5G/B5G: A Trust-Based Task Mapping Solution and Open Research Issues," *IEEE Access*, **8**, 117218–117235, doi: 10.1109/ACCESS.2020.3004738 (2020).

[12] W. Ben Jaballah, M. Conti, and C. Lal, "Security and design requirements for software-defined VANETs," *Comput. Networks*, **169**, 107099, doi: 10.1016/j.comnet.2020.107099 (2020).

[13] T. Mekki, I. Jabri, A. Rachedi, and L. Chaari, "Software-defined networking in vehicular networks: A survey," *Trans. Emerg. Telecommun. Technol.* **33**(10), 1–29, doi: 10.1002/ett.4265 (2021).

[14] A. Alalewi, I. Dayoub, and S. Cherkaoui, "On 5G-V2X Use Cases and Enabling Technologies: A Comprehensive Survey," *IEEE Access* **9**, 107710–107737, doi: 10.1109/ACCESS.2021.3100472 (2021).

[15] N. Cardona, E. Coronado, S. Latre, R. Riggio, and J. M. Marquez-Barja, "Software-Defined Vehicular Networking: Opportunities and Challenges," *IEEE Access*, **8**(February 2021), 219971–219995, doi: 10.1109/ACCESS.2020.3042717 (2020).

[16] N. H. Hussein, C. T. Yaw, S. P. Koh, S. K. Tiong, and K. H. Chong, "A Comprehensive Survey on Vehicular Networking: Communications, Applications, Challenges, and Upcoming Research Directions," *IEEE Access* **10**(August 2022), 86127–86180, doi: 10.1109/ACCESS.2022.3198656 (2022).

[17] N. Phull and P. Singh, "A review on security issues in VANETs," *Proc. 2019 6th Int. Conf. Comput. Sustain. Glob. Dev. INDIACom* **2019**, 1084–1088, (2019).

[18] S. A. Jan, N. U. Amin, M. Othman, M. Ali, A. I. Umar, and A. Basir, "A Survey on Privacy-Preserving Authentication Schemes in VANETs: Attacks, Challenges and Open Issues," *IEEE Access* **9**, 153701–153726, doi: 10.1109/ACCESS.2021.3125521 (2021).

[19] Z. Lu, G. Qu, and Z. Liu, "A Survey on Recent Advances in Vehicular Network Security, Trust, and Privacy," *IEEE Trans. Intell. Transp. Syst.* **20**(2), 760–776, doi: 10.1109/TITS.2018.2818888 (2019).

[20] T. Pavithra and B. S. Nagabhushana, "A Survey on Security in VANETs," *Proc. 2nd Int. Conf. Inven. Res. Comput. Appl. ICIRCA* ***2020***, 881–889, doi: 10.1109/ICIRCA48905.2020.9182823 (2020).

[21] H. Amari, Z. A. El Houda, L. Khoukhi, and L. H. Belguith, "Trust Management in Vehicular Ad-Hoc Networks: Extensive Survey," *IEEE Access* **11**(March), 47659–47680, doi: 10.1109/ACCESS.2023.3268991 (2023).

[22] R. Hussain, F. Hussain, S. Zeadally, and J. Y. Lee, "On the Adequacy of 5G Security for Vehicular Ad Hoc Networks," *IEEE Commun. Stand. Mag.* **5**(1), 32–39, doi: 10.1109/MCOMSTD.001.2000066 (2021).

[23] M. A. Al-Shareeda, M. Anbar, S. Manickam, A. Khalil, and I. H. Hasbullah, "Security and Privacy Schemes in Vehicular Ad-Hoc Network with Identity-Based Cryptography Approach: A Survey," *IEEE Access* **9**, 121522–121531, doi: 10.1109/ACCESS.2021.3109264 (2021).

[24] D. Zhang, F. R. Yu, R. Yang, and L. Zhu, "Software-Defined Vehicular Networks with Trust Management: A Deep Reinforcement Learning Approach," *IEEE Trans. Intell. Transp. Syst.* **23**(2), 1400–1414, doi: 10.1109/TITS.2020.3025684 (2022).

[25] M. Arif, G. Wang, M. Zakirul Alam Bhuiyan, T. Wang, and J. Chen, "A survey on security attacks in VANETs: Communication, applications and challenges," *Veh. Commun.* **19**, 100179, doi: 10.1016/j.vehcom.2019.100179 (2019).

[26] S. Babu, I. Ghosh, and B. S. Manoj, "Effort: A New Metric for Roadside Unit Placement in 5G Enabled Vehicular Networks," *2020 IEEE 3rd 5G World Forum, 5GWF 2020 - Conf. Proc.*, 263–268, doi: 10.1109/5GWF49715.2020.9221228 (2020).

[27] M. J. N. Mahi *et al.*, "A Review on VANET Research: Perspective of Recent Emerging Technologies," *IEEE Access* **10**, 65760–65783, doi: 10.1109/ACCESS.2022.3183605 (2022).

[28] J. Gao *et al.*, "A Blockchain-SDN-Enabled Internet of Vehicles Environment for Fog Computing and 5G Networks," *IEEE Internet Things J.* **7**(5), 4278–4291, doi: 10.1109/JIOT.2019.2956241(2020).

[29] R. Kumar and N. Agrawal, *A survey on software-defined vehicular networks (SDVNs): a security perspective* **79**(8), doi: 10.1007/s11227-022-05008-y (Springer US, 2023).

[30] I. M. Varma and N. Kumar, "A comprehensive survey on SDN and blockchain-based secure vehicular networks," *Veh. Commun.* **44**, 100663, doi: 10.1016/j.vehcom.2023.100663 (2023).

[31] [31] D. Tian *et al.*, "A microbial inspired routing protocol for VANETs," *IEEE Internet Things J.*, vol. 5, no. 4, pp. 2293–2303, 2018, doi: 10.1109/JIOT.2017.2737466.

[32] J. He, L. Cai, J. Pan, and P. Cheng, "Delay Analysis and Routing for Two-Dimensional VANETs Using Carry-and-Forward Mechanism," *IEEE Trans. Mob. Comput.* **16**(7), 1830–1841, doi: 10.1109/TMC.2016.2607748 (2017).

[33] M. M. Islam, M. T. R. Khan, M. M. Saad, and D. Kim, "Software-defined vehicular network (SDVN): A survey on architecture and routing," *J. Syst. Archit.* **114**(September 2020), 101961, doi: 10.1016/j.sysarc.2020.101961 (2021).

[34] J. S. Weng, J. Weng, Y. Zhang, W. Luo, and W. Lan, "BENBI: Scalable and dynamic access control on the northbound interface of SDN-Based VANET," *IEEE Trans. Veh. Technol.* **68**(1), 822–831, doi: 10.1109/TVT.2018.2880238 (2019).

[35] L. F. Eliyan and R. Di Pietro, "DoS and DDoS attacks in Software Defined Networks: A survey of existing solutions and research challenges," *Futur. Gener. Comput. Syst.* **122**, 149–171, doi: 10.1016/j.future.2021.03.011 (2021).

[36] A. S. Alqahtani, "Security threats and countermeasures in software defined network using efficient and secure trusted routing mechanism," *Comput. Commun.* **153**(January), 336–341, doi: 10.1016/j.comcom.2020.02.020 (2020).

[37] S. Lee *et al.*, "A comprehensive security assessment framework for software-defined networks," *Comput. Secur.* **91**, 101720, doi: 10.1016/j.cose.2020.101720 (2020).

[38] R. Deb and S. Roy, "A Software Defined Network information security risk assessment based on Pythagorean fuzzy sets," *Expert Syst. Appl.* **183**, (December 2020), 115383, doi: 10.1016/j.eswa.2021.115383 (2021).

[39] R. ur Rasool, H. Wang, U. Ashraf, K. Ahmed, Z. Anwar, and W. Rafique, "A survey of link flooding attacks in software defined network ecosystems," *J. Netw. Comput. Appl.* **172**, (March), 102803, doi: 10.1016/j.jnca.2020.102803 (2020).

[40] M. Priyadarsini and P. Bera, "Software defined networking architecture, traffic management, security, and placement: A survey," *Comput. Networks* **192**, (June 2020), 108047, doi: 10.1016/j.comnet.2021.108047 (2021).

[41] B. Yan, Q. Liu, J. Shen, D. Liang, B. Zhao, and L. Ouyang, "A survey of low-latency transmission strategies in software defined networking," *Comput. Sci. Rev.*, **40**, 100386, doi: 10.1016/j.cosrev.2021.100386 (2021).

[42] I. A. Valdovinos, J. A. Pérez-Díaz, K. K. R. Choo, and J. F. Botero, "Emerging DDoS attack detection and mitigation strategies in

software-defined networks: Taxonomy, challenges and future directions," *J. Netw. Comput. Appl.* **187**(June 2020), 103093, doi: 10.1016/j.jnca.2021.103093 (2021).

[43] R. Amin, I. Pali, and V. Sureshkumar, "Software-Defined Network enabled Vehicle to Vehicle secured data transmission protocol in VANETs," *J. Inf. Secur. Appl.* **58** (February), 102729, doi: 10.1016/j.jisa.2020.102729 (2021).

[44] K. Nisar *et al.*, "A survey on the architecture, application, and security of software defined networking: Challenges and open issues," *Internet of Things (Netherlands)* **12**, 100289, doi: 10.1016/j.iot.2020.100289 (2020).

[45] Y. Liu, B. Zhao, P. Zhao, P. Fan, and H. Liu, "A survey: Typical security issues of software-defined networking," *China Commun.*, **16**(7), 13–31, doi: 10.23919/j.cc.2019.07.002 (2019).

[46] M. S. Farooq, S. Riaz, and A. Alvi, "Security and Privacy Issues in Software-Defined Networking (SDN): A Systematic Literature Review," *Electron.* **12**(14), doi: 10.3390/electronics12143077 (2023).

[47] L. Der Chou *et al.*, "Behavior Anomaly Detection in SDN Control Plane: A Case Study of Topology Discovery Attacks," *Wirel. Commun. Mob. Comput.* **2020**, 357–362, doi: 10.1155/2020/8898949 (2020).

[48] A. Di Maio *et al.*, "Enabling SDN in VANETs: What is the impact on security?," *Sensors (Switzerland)* **16**(12), 1–24, doi: 10.3390/s16122077 (2016).

[49] Z. Yu, H. Zhu, R. Xiao, C. Song, J. Dong, and H. Li, "Detection and defense against network isolation attacks in software-defined networks," *Trans. Emerg. Telecommun. Technol.* **32**(5), 1–16, doi: 10.1002/ett.3895 (2021).

[50] E. Calle, D. Martínez, M. Mycek, and M. Pióro, "Resilient backup controller placement in distributed SDN under critical targeted attacks," *Int. J. Crit. Infrastruct. Prot.* **33**, 100422, doi: 10.1016/j.ijcip.2021.100422 (2021).

[51] K. Y. Chen *et al.*, "SDNShield: NFV-Based Defense Framework Against DDoS Attacks on SDN Control Plane," *IEEE/ACM Trans. Netw.* **30**(1), 1–17, doi: 10.1109/TNET.2021.3105187 (2022).

[52] R. Xie *et al.*, "Disrupting the SDN Control Channel via Shared Links: Attacks and Countermeasures," *IEEE/ACM Trans. Netw.* **30**(5), 2158–2172, doi: 10.1109/TNET.2022.3169136 (2022).

[53] M. Lacoste, D. Armand, F. L'hereec, F. Prévost, Y. Rafflé, and S. Roché, "Software-Defined Vehicular Networking Security: Threats and Security Opportunities for 5G," *https://www.cesar-conference.org/wp-content/uploads/2019/10/20191119_J1_090_M-LACOSTE_Software_Defined_Vehicular_Network.pdf*, 2020, [Online]. Available: https://www.cesar-conference.org/wp-content/uploads/2019/10/20191119_J1_090_M-LACOSTE_Software_Defined_Vehicular_Network.pdf

[54] N. Noorani and S. A. H. Seno, "Routing in VANETs based on intersection using SDN and fog computing," *2018 8th Int. Conf. Comput. Knowl. Eng. ICCKE 2018*, no. Iccke, 339–344, doi: 10.1109/ICCKE.2018.8566352 (2018).

[55] K. S. Kalupahana Liyanage, M. Ma, and P. H. Joo Chong, "Controller placement optimization in hierarchical distributed software defined vehicular networks," *Comput. Networks*, vol. **135**, 226–239, doi: 10.1016/j.comnet.2018.02.022 (2018).

[56] S. Toufga, S. Abdellatif, H. T. Assouane, P. Owezarski, and T. Villemur, "Towards dynamic controller placement in software defined vehicular networks," *Sensors (Switzerland)* **20**(6), 1–20, doi: 10.3390/s20061701 (2020).

[57] S. Hakak *et al.*, "Autonomous vehicles in 5G and beyond: A survey," *Veh. Commun.* **39**, 1–34, doi: 10.1016/j.vehcom.2022.100551 (2022).

[58] R. Hussain, "Integration of VANET and 5G Security: A review of design and implementation issues," *Futur. Gener. Comput. Syst.*, **101**, 843–864, doi: 10.1016/j.future.2019.07.006 (2019).

[59] H. Amari, Z. A. El Houda, L. Khoukhi, and L. H. Belguith, "Trust Management in Vehicular Ad-Hoc Networks: Extensive Survey," *IEEE Access* **11**(May), 47659–47680, doi: 10.1109/ACCESS.2023.3268991 (2023).

[60] S. Khan, A. Gani, A. W. Abdul Wahab, M. Guizani, and M. K. Khan, "Topology Discovery in Software Defined Networks: Threats, Taxonomy, and State-of-the-Art," *IEEE Commun. Surv. Tutorials* **19**(1), 303–324, doi: 10.1109/COMST.2016.2597193 (2017).

[61] S. Soltani, A. Amanlou, M. Shojafar, and R. Tafazolli, "Security of Topology Discovery Service in SDN: Vulnerabilities and Countermeasures," *IEEE Open J. Commun. Soc.* **5**(June), 3410–3450, doi: 10.1109/OJCOMS.2024.3406489 (2024).

[62] A. Sarkunavathi and V. Srinivasan, "A Scrutinized study on DoS attacks in Wireless Sensor Networks and need of SDN in Mitigating DoS attacks," *2021 Int. Conf. Comput. Commun. Informatics, ICCCI 2021*, 2021, doi: 10.1109/ICCCI50826.2021.9402459.

[63] A. Al Hayajneh, M. Z. A. Bhuiyan, and I. McAndrew, "Improving internet of things (IoT) security with software-defined networking (SDN)," *Computers* **9**(1), 1–14, doi: 10.3390/computers9010008 (2020).

[64] A. Pradhan and R. Mathew, "Solutions to Vulnerabilities and Threats in Software Defined Networking (SDN)," *Procedia Comput. Sci.* **171**(2019), 2581–2589, doi: 10.1016/j.procs.2020.04.280 (2020).

[65] R. Gonzaga, P. Nazareno Maia Sampaio, R. Gonzaga Silva, and P. Maia Sampaio Nazareno, "Mitigating Man In The Middle attacks within Context-based SDNs," 1–8, 2020, [Online]. Available: https://hal.science/hal-02495155

[66] A. Rahman *et al.*, "DistB-SDoIndustry: Enhancing security in industry 4.0 services based on distributed blockchain through software defined networking-IoT enabled architecture," *Int. J. Adv. Comput. Sci. Appl.*, **11**(9), 674–681, doi: 10.14569/IJACSA.2020.0110980 (2020).

[67] W. Iqbal, H. Abbas, B. Rauf, Y. A. Bangash, M. F. Amjad, and A. Hemani, "PCSS: Privacy Preserving Communication Scheme for SDN Enabled Smart Homes," *IEEE Sens. J.* **22**(18), 17677–17690, doi: 10.1109/JSEN.2021.3087779 (2022).

[68] E. P. Neto, F. S. D. Silva, L. M. Schneider, A. V. Neto, and R. Immich, "Seamless MANO of multi-vendor SDN controllers across federated multi-domains," *Comput. Networks* **186**(July), 107752, 2021, doi: 10.1016/j.comnet.2020.107752 (2020).

[69] J. C. Correa Chica, J. C. Imbachi, and J. F. Botero Vega, "Security in SDN: A comprehensive survey," *J. Netw. Comput. Appl.* **159**, (November), 102595, 2020, doi: 10.1016/j.jnca.2020.102595 (2019).

[70] S. Madhawa, P. Balakrishnan, and U. Arumugam, "Roll forward validation based decision tree classification for detecting data integrity attacks in industrial internet of things," *J. Intell. Fuzzy Syst.* **36**(3), 2355–2366, doi: 10.3233/JIFS-169946 (2019).

[71] M. Karimi and P. Krishnamurthy, "Software defined ambit of data integrity for the internet of things," *Proc. - 21st IEEE/ACM Int.*

Symp. Clust. Cloud Internet Comput. CCGrid 2021, 737–745, doi: 10.1109/CCGrid51090.2021.00089 (2021).

[72] P. T. Duy, H. Do Hoang, D. T. Thu Hien, N. Ba Khanh, and V. H. Pham, "SDNLog-Foren: Ensuring the integrity and tamper resistance of log files for SDN forensics using blockchain," *Proc. - 2019 6th NAFOSTED Conf. Inf. Comput. Sci. NICS 2019*, 416–421, doi: 10.1109/NICS48868.2019.9023852 (2019).

[73] W. Iqbal *et al.*, "ALAM: Anonymous Lightweight Authentication Mechanism for SDN-Enabled Smart Homes," *IEEE Internet Things J.* **8**(12), 9622–9633, doi: 10.1109/JIOT.2020.3024058 (2021).

[74] J. Cao, M. Ma, Y. Fu, H. Li, and Y. Zhang, "CPPHA: Capability-Based Privacy-Protection Handover Authentication Mechanism for SDN-Based 5G HetNets," *IEEE Trans. Dependable Secur. Comput.* **18**(3), 1182–1195, doi: 10.1109/TDSC.2019.2916593 (2021).

[75] A. H. Abdi *et al.*, "Security Control and Data Planes of SDN: A Comprehensive Review of Traditional, AI, and MTD Approaches to Security Solutions," *IEEE Access* **12**(April), 69941–69980, doi: 10.1109/ACCESS.2024.3393548 (2024).

[76] X. Liang and H. Chen, "A SDN-Based hierarchical authentication mechanism for IPv6 address," *2019 IEEE Int. Conf. Intell. Secur. Informatics, ISI 2019*, 225, doi: 10.1109/ISI.2019.8823463 (2019).

[77] L. Fang *et al.*, "THP: A Novel Authentication Scheme to Prevent Multiple Attacks in SDN-Based IoT Network," *IEEE Internet Things J.* **7**(7), 5745–5759, doi: 10.1109/JIOT.2019.2944301 (2020).

[78] K. Deepthika, T. Vanaja, S. Keerthika, and S. N. Prajwalasimha, "AI-Enabled DDoS Detection and Mitigation in the Software Defined Network," *5th Int. Conf. Electron. Sustain. Commun. Syst. ICESC 2024 - Proc.*, no. Icesc, 663–667, doi: 10.1109/ICESC60852.2024.10689743 (2024).

[79] F. A. Munmun and M. Paul, "Challenges of DDoS Attack Mitigation in IoT Devices by Software Defined Networking (SDN)," *2021 Int. Conf. Sci. Contemp. Technol. ICSCT 2021*, 1–5, doi: 10.1109/ICSCT53883.2021.9642640 (2021).

[80] B. T. Alemu and A. J. Muhammed, "Controller-Targeted DDoS Attack Detection and Mitigation in Software-Defined Internet of Vehicles (SD-IoV)," *2023 Int. Conf. Inf. Commun. Technol. Dev. Africa, ICT4DA 2023*, 138–143, doi: 10.1109/ICT4DA59526.2023.10302231 (2023).

[81] H. S. Abdulkarem and A. Dawod, "DDoS Attack Detection and Mitigation at SDN Data Plane Layer," *Proc. - 2020 IEEE 2nd Glob. Power, Energy Commun. Conf. GPECOM 2020*, 322–326, doi: 10.1109/GPECOM49333.2020.9247850 (2020).

[82] C. Kaushik, "DDoS ATTACK DETECTION AND MITIGATION USING MININET AND RYU CONTROLLER IN SDN ENVIRONMENT," doi: 10.1109/ICCCNT61001.2024.10724700 (2024).

[83] A. N. H. Dhatreesh Sai, B. H. Tilak, N. Sai Sanjith, P. Suhas, and R. Sanjeetha, "Detection and Mitigation of Low and Slow DDoS attack in an SDN environment," *2022 IEEE Int. Conf. Distrib. Comput. VLSI, Electr. Circuits Robot. Discov. 2022 - Proc.*, 106–111, doi: 10.1109/DISCOVER55800.2022.9974724 (2022).

[84] N. Aslam, S. Srivastava, and M. M. Gore, "Evaluating DDoS Detection and Mitigation in SDN at Various Attack Rates," *1st Int. Conf. Pioneer. Dev. Comput. Sci. Digit. Technol. IC2SDT 2024 - Proc.*, 569–574, doi: 10.1109/IC2SDT62152.2024.10696232 (2024).

[85] G. De Biasi, L. F. M. Vieira, and A. A. F. Loureiro, "Sentinel: Defense mechanism against DDoS flooding attack in software defined vehicular network," *IEEE Int. Conf. Commun.*, **2018**(May), doi: 10.1109/ICC.2018.8422303 (2018).

[86] H. Li, F. Wei, and H. Hu, "Enabling dynamic network access control with anomaly-based IDS and SDN," *SDN-NFV 2019 - Proc. ACM Int. Work. Secur. Softw. Defin. Networks Netw. Funct. Virtualization, co-located with CODASPY 2019*, 13–16, doi: 10.1145/3309194.3309199 (2019).

[87] M. Cheminod, L. Durante, L. Seno, F. Valenza, and A. Valenzano, "A comprehensive approach to the automatic refinement and verification of access control policies," *Comput. Secur.* **80**, 186–199, doi: 10.1016/j.cose.2018.09.013 (2019).

[88] I. H. Abdulqadder, S. Zhou, I. T. Aziz, D. Zou, X. Deng, and S. M. Abrar Akber, "An Effective Lightweight Intrusion Detection System with Blockchain to Mitigate Attacks in SDN/NFV Enabled Cloud," *2021 6th Int. Conf. Converg. Technol. I2CT 2021*, doi: 10.1109/I2CT51068.2021.9417961 (2021).

[89] J. S. Weng, J. Weng, Y. Zhang, W. Luo, and W. Lan, "BENBI: Scalable and dynamic access control on the northbound interface of SDN-Based VANET," *IEEE Trans. Veh. Technol.* **68**(1), 822–831, doi: 10.1109/TVT.2018.2880238 (2019).

[90] J. Ramprasath and V. Seethalakshmi, "Secure access of resources in software-defined networks using dynamic access control list," *Int. J. Commun. Syst.* **34**(1), 1–12, doi: 10.1002/dac.4607 (2021).

[91] F. Miri and R. Pazzi, "A Comprehensive Survey on the Convergence of Vehicular Social Networks and Fog Computing," 2021, [Online]. Available: http://arxiv.org/abs/2112.00143

[92] Y. He *et al.*, "D2D-V2X-SDN: Taxonomy and Architecture towards 5G Mobile Communication System," *IEEE Access* **9**, 155507–155525, doi: 10.1109/ACCESS.2021.3127041 (2021).

[93] D. Tatang, F. Quinkert, J. Frank, C. Röpke, and T. Holz, "SDN-GUARD: Protecting SDN controllers against SDN rootkits," *2017 IEEE Conf. Netw. Funct. Virtualization Softw. Defin. Networks, NFV-SDN 2017* **2017**(Janua), 297–302, doi: 10.1109/NFV-SDN.2017.8169856 (2017).

[94] R. Mohammadi, R. Javidan, and M. Conti, "SLICOTS: An SDN-based lightweight countermeasure for TCP SYN flooding attacks," *IEEE Trans. Netw. Serv. Manag.* **14**(2), 487–497, doi: 10.1109/TNSM.2017.2701549 (2017).

[95] H. Wang, L. Xu, and G. Gu, "FloodGuard: A DoS Attack Prevention Extension in Software-Defined Networks," *Proc. Int. Conf. Dependable Syst. Networks* **2015**(Septe), 239–250, doi: 10.1109/DSN.2015.27 (2015).

[96] M. Ambrosin, M. Conti, F. De Gaspari, and R. Poovendran, "LineSwitch: Tackling Control Plane Saturation Attacks in Software-Defined Networking," *IEEE/ACM Trans. Netw.* **25**(2), 1206–1219, doi: 10.1109/TNET.2016.2626287 (2017).

[97] Y. Wang, T. Hu, G. Tang, J. Xie, and J. Lu, "SGS: Safe-Guard Scheme for Protecting Control Plane Against DDoS Attacks in Software-Defined Networking," *IEEE Access* **7**(c), 34699–34710, doi: 10.1109/ACCESS.2019.2895092 (2019).

[98] F. Ruffy, W. Hommel, and F. von Eye, "A STRIDE-based security architecture for software-defined networking. ICN 2016, p.107.," *A STRIDE-based Secur. Archit. software-defined Netw.*, (c), 107, 2016.

[99] R. Sahay, G. Blanc, Z. Zhang, and H. Debar, "ArOMA: An SDN based autonomic DDoS mitigation framework," *Comput. Secur.* **70**, 482–499, doi: 10.1016/j.cose.2017.07.008 (2017).

[100] V. K. Tchendji, F. Mvah, C. T. Djamegni, and Y. F. Yankam, "E2BaSeP: Efficient Bayes Based Security Protocol Against ARP

Spoofing Attacks in SDN Architectures," *J. Hardw. Syst. Secur.***5**(1), 58–74, doi: 10.1007/s41635-020-00105-x (2021).

[101] N. Ahmed *et al.*, "Network Threat Detection Using Machine/Deep Learning in SDN-Based Platforms: A Comprehensive Analysis of State-of-the-Art Solutions, Discussion, Challenges, and Future Research Direction," *Sensors (Basel).* **22**(20), doi: 10.3390/s22207896 (2022).

[102] M. U. Ghazi, M. A. Khan Khattak, B. Shabir, A. W. Malik, and M. Sher Ramzan, "Emergency Message Dissemination in Vehicular Networks: A Review," *IEEE Access* **8**, 38606–38621, doi: 10.1109/ACCESS.2020.2975110 (2020).