A Review on software defined vehicle

Pradeep Nayak*[1], Vithika Shetty*[2] , Syed Saleha*[3], Suvarna Arvinkanth*[4], Suvan P Kedilnaya*[5], Vasavi Rai C*[6]

Alvas Institute of Engineering and Technology, Mijar, Karnataka, India- 574225.

Department of Information Science and Engineering.

Abstract

The automobile industry is being completely transformed by software-defined vehicles (SDVs), which offer unprecedented flexibility, adaptability, and connectivity. This study provides a thorough analysis of SDVs, outlining their fundamental ideas, major technologies, difficulties, current fixes, and prospective prospects. We examine recent commercial developments and cutting-edge research to shed light on the possible advantages and dangers of SDVs. We also point out significant research gaps and suggest intriguing lines of inquiry for ongoing work in this fascinating area.

A paradigm shift in the automobile industry has been made possible by the rise of software-defined vehicles, which have transformed conventional vehicles into intelligent, networked, and highly adjustable modes of transportation. This abstract gives a general introduction of software-defined cars and emphasises how they could completely change how people travel in the future.

Modern technology and sophisticated software are used in software-defined vehicles to improve user experience, performance, and usefulness. These vehicles provide unparalleled degrees of flexibility, adaptability, and upgradeability by utilising sophisticated computing capabilities. Manufacturers may continuously improve and add new features with over-the-air updates, ensuring that vehicles keep up with the most recent developments in technology.

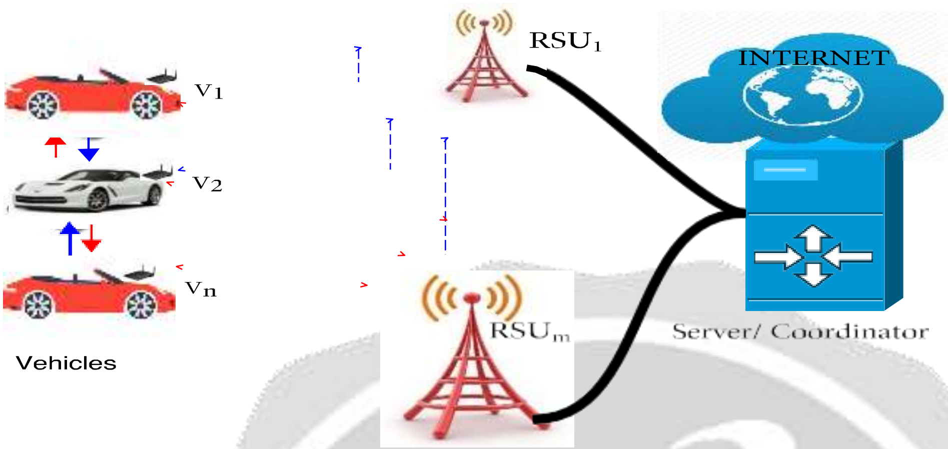
To obtain information about the environment in real-time, these vehicles rely on a wide range of sensors, cameras, and communication technologies. Software-defined vehicles can use this data to analyse decisions, increase safety, and enhance driving efficiency. They do this by using artificial intelligence and machine learning algorithms. The overall mobility experience is further improved by their seamless connection with other smart devices and systems, including smart homes, smartphones, and intelligent transportation networks.

Key words

Network architecture, heterogeneous network, resource utilisation effectiveness, vehicle-to-infrastructure, and software defined networking are some terms used to describe the Internet of Vehicles.

Introduction

It is anticipated that the world will soon be filled with sentient items as a result of the beginning of a revolutionary era. In particular, thanks to the extraordinary advancement of enabling technologies in IT (Information Technology), smart devices that are part of our daily lives are now equipped with communication capabilities via wireless and backbone networks in addition to computation capabilities via embedded chips and remote clouds. Such a new paradigm, more often known as loT (Internet of Things)[11, will significantly improve our way of living in the future. The 10V (Internet of Vehicles)[21, an essential component and pioneer of the enormous loT family, has attracted significant interest from the automotive and information technology groups and has advanced technologically. According to the ITS (Intelligent Transportation System)[3] plan, which is intended to provide a range of services from mobile Internet connectivity to traffic safety. For instance, in the event of a traffic accident, information would be broadcast via V2V communications to all nearby vehicles, and drivers heading in that direction would be given alternate routes. The cloud-based trip planner programme may also decide what is best for each driver who uses the service by gathering and analysing real-time traffic data from vehicles and sensors placed alongside the roadways. The term "software-defined vehicle" refers to a vehicle whose features and operations are essentially made possible by software. This continual change of the automobile from a mostly hardware-based product to an electronic gadget on wheels with software at its core has led to this development. Second, it presents a considerable difficulty to meet the various QoS (Quality of Service) requirements in VCNs. Future connected cars are anticipated to play a significant role as mobile information gathering and processing hubs in everyday life.

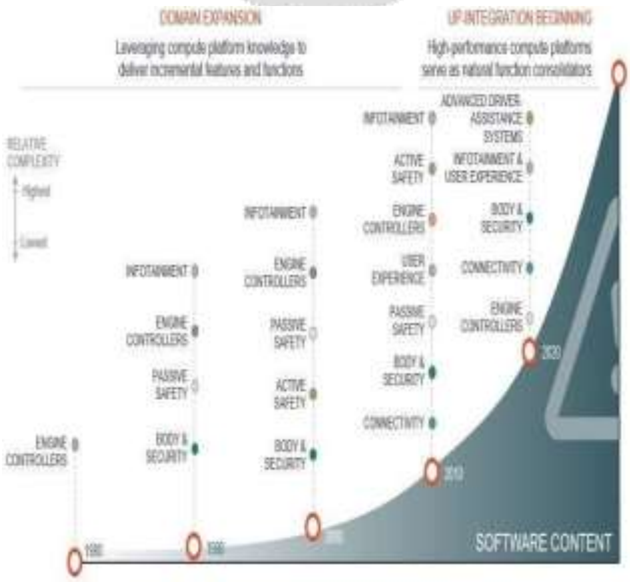


Therefore, IoV ought to be able to support requests for differentiated services with various QoS requirements. Safety-related services, for instance, demand low latency and high reliability, streaming services place stringent restrictions on connection speed and stability, and delay-tolerant services typically use a lot of bandwidth. The distinction of these services, however, is barely supported by diverse network substrates, and the network lacks a comprehensive understanding of all service requests, making it difficult to reach agreements among them in order to satisfy as many requests as feasible.

Software defined IoV: architecture

SDN logical controllers. SDN controllers are essentially server-based software programmes that have a comprehensive view of every other component in the SD-IoV system. As the fundamental building blocks of SD-IoV, they are in charge of both basic network management and operation (such as rule generation, network virtualization, client association, resource allocation, and mobility management) as well as some more sophisticated functionalities (such as data pre-processing, network analysis, and learning). The term "logical" has two different meanings. On the one hand, this suggests that SDN controllers may be physically installed in local or cloud environments (for scalability and delay-reduction purposes). On the other hand, all of these functions can be spread hierarchically across various controllers.

- Electronic Control Units (ECUs): In software-defined vehicles, ECUs act as the core processing units. These tiny computers manage and operate several automotive subsystems, including communication,



braking, steering, and engine control.

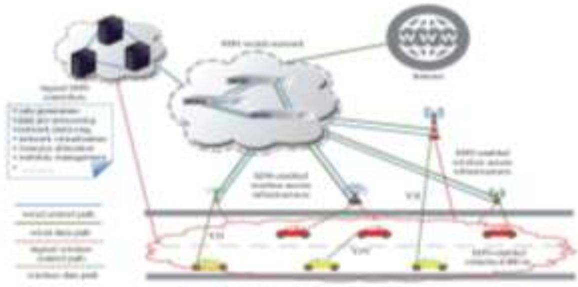


Figure 1 Architecture of software defined IoV

- Software-defined cars rely on a wide range of sensors to collect information about their environment. These sensors come in a variety of forms, such as cameras, lidar, radar, ultrasonic sensors, and others. For perception, object detection, and situational awareness, they offer essential information.
- Connectivity: To facilitate connection with external systems and devices, software-defined
- cars heavily rely on connectivity technologies, such as cellular networks, Wi-Fi, and Bluetooth. Over-the-air updates, real-time data transfer, and integration with intelligent transportation systems are all made possible by this connectivity.
- Centralised Computing Platform: The software-defined vehicle's brain is a centralised computing platform. It is made up of strong CPUs, lots of storage, and memory resources. This platform administers software programmes operating on the vehicle, processes sensor data, and runs sophisticated algorithms for decision-making.
- Software Frameworks: Within the architecture of the vehicle, software frameworks serve as the building blocks for the development and integration of applications. These frameworks make it easier to deploy software modules for a variety of purposes, including connectivity, driver assistance, autonomous driving, and infotainment.
- Human-Machine Interface (HMI): This element permits communication between the driver and the computerised systems of the vehicle. It incorporates interfaces like speech recognition, steering wheel controls, touchscreens, and augmented reality displays. The HMI enables users to access car features, manage settings, and get software feedback.
- Software complexity and increased connection make cybersecurity a key component of software-defined vehicle design. To guard against potential cyber attacks, strong security measures are put in place, such as secure communication protocols, encryption methods, intrusion detection systems, and over-the-air security updates.
- Cloud Integration: To store, analyse, and analyse data, software-defined vehicles frequently use cloud-based services. Access to advanced analytics, machine learning algorithms, and remote diagnostics is made possible through cloud integration, which improves vehicle performance and makes it possible to provide individualised services.
- route in control.:The primary duties of control paths are to 1) ensure that the other SD-IoV components may be managed by the controllers, and 2) ensure real-time status feedbacks from other components. Similar to conventional SDNs, wired control channels are employed between SDN controllers, SDN switch networks, and SDN-enabled wireless access infrastructures. Although there are various approaches to implement wireless control pathways and various control channel needs, hardware and protocol support. As a result, in the example architecture, we only employ logical wireless control pathways to connect the vehicles and controllers.

  In order to more clearly demonstrate the SD10V from a new perspective, we show its layered design in Fig. 2.Below, these planes are introduced from top to bottom.

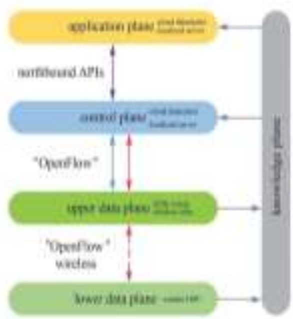


Figure 2 Layered architecture of software defined IoV

Vehicular network virtualization

The SD-IoV is by no means an exception to the rule that SDN is inherently suitable for achieving network virtualization. To be more sp network resources can be easily divided and distinctly isolated through the control of traffic flow and design of wireless access infrastructures as well as vehicles, which is referred to as network virtualization. It appears as though a certain user or service is the only owner of the entire network from their point of view. Since multiple users can share physical infrastructures (like switches), the first goal of network virtualization is to increase resource utilisation efficiency.by creating a virtualized network with specific resources for the services, it may also be utilised to ensure the QoS of high-priority services in IoV.

Handover of vehicular access

When mobile users enter the intersection region of adjacent cells in typical wireless networks, the handoff procedure is passively activated. The primary cause of performance decline. However, because the controller has a global perspective in SD-IoV, the handoff process can be prepared ahead of time. Furthermore, the handoff procedure itself can be greatly simplified. For example, for each service/vehicle, a virtualized agent can be created that stores the essential information, such as the destination IP address, to keep the connection open. During a handoff, the controller simply migrate

s the virtualized agent from the original infrastructure to the new one while allocating network resources and informing the handoff vehicle of upcoming activities such as interface changes.

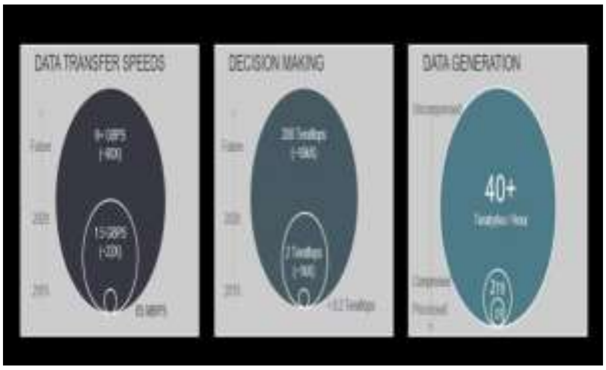
The Present-Future Gap

- The automotive hardware and software architecture does not support software-defined cars. The electrical and electronic design has revealed computational power restrictions and communication difficulties. efficiency, and uncontrolled expenses of wire harness.
- The typical waterfall software development model is severely limited. Based on the aforementioned changes in technology architecture, automobile R&D will shift from the traditional waterfall development paradigm to an agile development strategy in the context of software-defined vehicles.
- The key shortcomings of software-oriented automotive transformation are organisational structure and talent supply.The organisational structure of OEMs will radically change, shifting from a function-oriented structure to a platform development structure.
- Supply chain system impediments. The link between vehicle and components firms shifts from a tower-shaped vertical relationship to an annular relationship.

The advantages of a 'Software-defined vehicle'

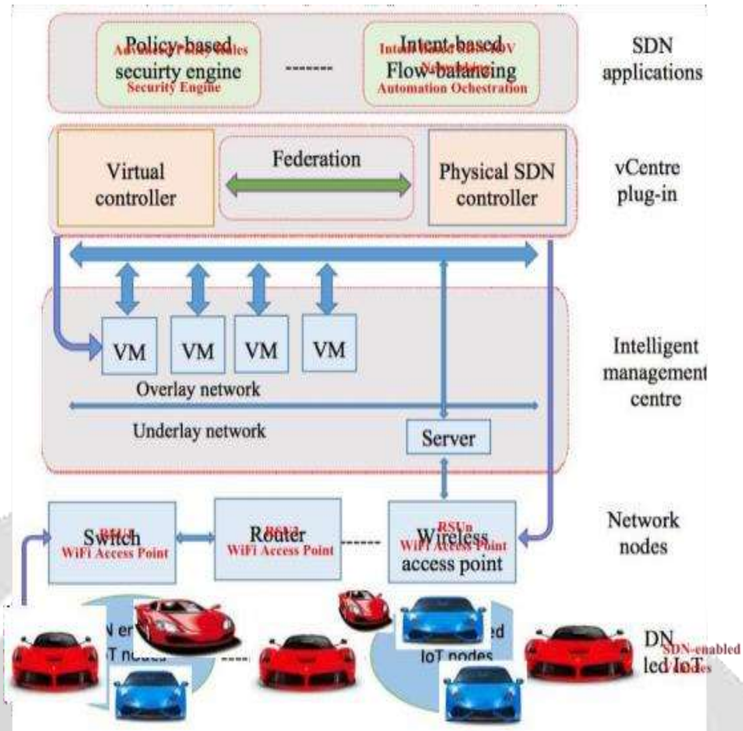
- Aside from enabling additional safety, comfort, and convenience features, the software-defined vehicle offers other advantages over its hardware-defined predecessor.
- Software changes to car infotainment, telematics, or diagnostic systems now necessitate a trip to the dealership. Customers who purchase a software-defined car will be able to receive over-the-air (OTA) upgrades that include security fixes, infotainment improvements, and monitoring and adjustment of core functioning capabilities. such as the powerplant and vehicle dynamics.
- ECUs will send and receive massive volumes of data to and from sensors and actuators, providing

vehicle manufacturers with unprecedented insight into every aspect of a vehicle, its performance, and its location in the connected ecosystem. This allows vehicle makers to improve life-cycle management.



Implementing SDVs presents challenges

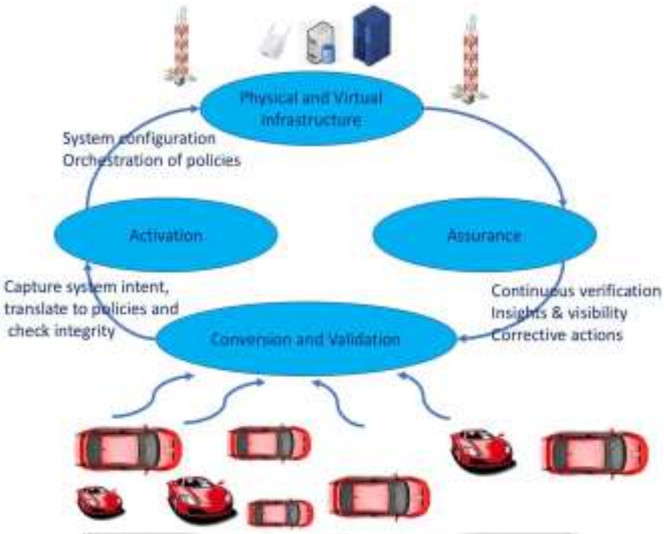
- **Safety and Reliability:** SDVs must adhere to strict safety regulations because safety is of utmost importance in the automotive sector. To avoid mishaps or malfunctions that can endanger passengers or other road users, it is essential to ensure the dependability and robustness of software systems, including the identification and mitigation of software problems.
- **Security:** SDVs are susceptible to cyberattacks due to their increasing connectedness and the presence of external communication ports. Critical hurdles in deploying SDV include preventing unauthorised access to the vehicle's software and data, establishing secure communication connections, and putting in place comprehensive cybersecurity safeguards.
- **Complex Software Integration:** SDVs contain several software components and subsystems, frequently created by various suppliers. It can be difficult to seamlessly integrate these components while retaining compatibility and interoperability; doing so calls for standard interfaces, protocols, and cooperation between various software modules.
- **Processing and analysis of data:** SDVs collect a tonne of data from a variety of sensors, cameras, and other devices. There are computational and algorithmic difficulties in effectively processing, analysing, and deriving relevant insights from this data in real-time. To address the large data volume and complexity, advanced data processing techniques like edge computing and distributed computing may be needed.
- **Regulation and Legal Considerations:** The use of SDVs poses regulatory and legal issues. Collaboration between industry stakeholders and politicians is necessary to adapt current legislation to account for the particular features of SDVs, such as liability problems in autonomous driving scenarios, data privacy difficulties, and ethical considerations. This is necessary to ensure that the right frameworks are in place.
- **Infrastructure Readiness:** SDVs are dependent on sophisticated infrastructure, including fast communication networks, reliable GPS systems, and intelligent transportation systems. For SDV to be widely used, it is crucial to ensure that the required infrastructure, including seamless connectivity and dependable communication, is in place.
- **User Acceptance and Adoption:** Persuading consumers to accept and believe in SDVs is a difficult task. For SDVs to be widely adopted, it is crucial to increase public acceptance, resolve worries about safety, dependability, and loss of control, and communicate clearly about their advantages and constraints.
- **Cost and affordability:** The adoption of SDVs necessitates large expenditures in infrastructure, software development, advanced hardware components, and research and development. Manufacturers face a problem in finding cost-effective solutions while upholding high standards, which could affect SDVs' accessibility and pricing.



Designed for Scalability

- Virtual overlay networks must be very compatible with physical underlay SDN-controlled IoV networks in such mobile data centres of cars. Even while the overlay does not give any information on time-varying changes in the underlying network, it results in high device delay since traffic patterns (and consequently network status) are rapidly changing in the underlying network of vehicles. In the SDN-IoV data networking, CF provides effective interoperability and unites virtual and physical resources.
- The CFs promise to offer a wide range of services in both underlay (physical) and overlay (virtual) networks. They primarily seek to increase access to multi-tenant hybrid networks and enhance current network infrastructures rather than replace them.
- Smart cities will make increased use of topological overlay control systems and other IoV network elements, which can quickly do a basic correlation examination to identify systems and facilities that are impacted by failures, overload, and congestion. Each vehicle may continue to operate with confidence, and by using clever criteria to evaluate the impact of changes to one layer (in the case of a loss, congestion, etc.), the overall interest can be increased. In essence, the federation will foster stronger mutual ties and high QoE. Additionally, it guarantees the interoperability of numerous IoV entities and vehicles.

In FIGS. 2 and 3, a single virtual, central entity (VMware NSX controller) manages a number of physically dispersed controllers (vehicles/VMs). The federation and suggestions from the HP and VMware companies improve communication between VMware and the hardware SDN controller [5].



Components of the SDN-IoV intent-based data networking for connected vehicles are shown in Figure 3.

Overall, employing the SDN-based CF, the overlay and underlay networks may cooperate successfully. The overlay network constantly receives assurances from the underlay network regarding: i) the various services it may provide; and ii) notification from the underlay network when the overlay may stop providing such services. It's important to remember that the overlay and underlay may have either a strong connection mechanism or a weak interaction mechanism, depending on the requirements.

First point: We have shown that the vehicle-based data layer is dispersed across several domains, and we have also employed overlay techniques to address scalability issues. Now that the issue of scalability has been resolved, we assume that every data flow in SDN-IoV networking must go through a thorough security inspection. The Smart Management System, which consists of the overlay network layer and is connected to a Policy Base Security Architecture (PbSA), which represents policies or rules, has also been presented.Each flow must sequentially move through the policy expression (rule) once it reaches PbSA in order to receive a response and be admitted into the SDN-IoV data networking system. A thorough investigation of the flow is performed following a suitable match of the

The security sub-module of SDNIoV data networking was thoroughly covered in the section after that, concentrating on the PbSA. Maximising traffic flows and implementing traffic diversion strategies based on the effective intent-based approach. In the paragraphs that follow, we'll talk about the importance of PbSA and our approach to intent-based traffic diverting.

Comprehension of SDN-IOV dynamics

This section acknowledges the potential for various network impairments in the various locations serviced by WiFi-based SDN-IoV data networking on the roads. We create a model to analyse the suggested framework and track the quantity of TCP packets passing through each vehicle's network system (recall that WiFi access points are shown in Fig. 1 as roadside units). Instead of relying on secure TCP connections, our experiment and numerical methodology uses a cross-layer architecture to measure the output and losses that SDN-enabled vehicles incur.

- **Security that is QoE-aware:** Using the proposed SDN-IoV framework, each vehicle will plug in and play with its own strategy and develop its own PbSA programming policy expressions to guarantee the necessary QoE service specifications.
- **QoE-aware Scalability:** By putting forth the SDN-IoV framework, we develop a useful module that enables a vehicle in the IoV network to regulate the dynamics of the data networking system by fine-tuning its own parameters and strengthening the IoV network system's resistance to shifting user numbers and channel dynamics. Clearly, under the federated control system, vehicles do offer options to cooperate with one another for scalability.

Software defined loV: open issues

Abstraction of resources: The complexity and intractability of fine-grained resource allocation are significantly increased by the heterogeneity of SD-10V, as was previously mentioned. The abstraction of physical resources is one potential solution to this issue. In other words, the formulation of the resource scheduling problem will be much clearer and the programmability provided by SDN will be used more effectively if the heterogeneous resources can be expressed in a unified language. The idea of resource virtualization has been recognised and used practically since the beginning of computer sciences.

Also, the use of multiplexing, such as TDM (Time Division Multiplexing), in wireless communications allows for the resource abstraction. The resources, however, are homogeneous in the circumstances mentioned above. Recent developments in LTE/4G networks, specifically OFDMA (Orthogonal Frequency-Division Multiple Access) technology, which uses resource blocks as the fundamental resource element, have resulted in a considerable advancement in resource abstraction. Even yet, the application of such abstraction to the heterogeneous SD-IoV may not be appropriate because it is still restricted to a single network. Building an abstraction layer between the wireless access infrastructures and the controller and defining the corresponding resource translation protocol is a potential first step in the right direction.

In order to deal with the service diversity and challenging environment of the 10V scenario, it is also important to study the evaluation method of the abstracted resources. As previously established, many resources with varying traits each have advantages and situations when they are useful. The evaluation process should be multi-dimensional in order to take into account these traits and, more significantly, to offer light on the best way to allocate these resources. update relevant information.

community intelligence: With SD-IoV, an intelligent network may be realised, and the controller can use learning-based algorithms to dynamically optimise the management of network resources. To thoroughly demonstrate this feature, we'll utilise two examples. A general cognitive cycle can be constructed to model the procedure in the first example. In addition to collecting real-time network state data, the controller also uses it as a stamp to store other data, like the resource utilisation index and the QoS satisfaction index.

The real SD10V can adopt the techniques after they are optimised and can surpass the current resource allocation algorithms (which are typically heuristic algorithms due to computation complexity). The real SD10V uses the real-time network status as input and outputs resource scheduling decisions. In the second illustration, prediction is accomplished using learning methodologies. In particular, the network density and mobility states for vehicles as well as the data traffic states are mined from previous data and used to make forecasts. Due to the close correlation between such state information and circumstances, such as time and location, in the 10V situation, it is quite regular. For instance, during rush hours, there are more automobiles on the road and they frequently request additional data when they are close to Pols (Points of Interest). Consequently, projections.

Privacy and security: The SD-IoV will face a number of security and privacy issues because it is a location-based and centrally controlled system. The controller needs circumstantial explanations of the state of all vehicles in the control plane, including their present locations and final destinations. Users of vehicles will face serious privacy difficulties if the controller cannot be trusted or if its databases are compromised. Additionally, SD10V is susceptible to DoS (Denial-of-Service) attacks as a result of centralization. A large number of fictitious requests sent in a brief period of time may cause the control functionality to become unresponsive. Additionally, if the controller is taken by unauthorised individuals, the attacker may gain control of the entire network and use it for other harmful activities, such as launching distributed denial-of-service attacks against other networks or important information.

Conclusion

In this study, we outlined the main obstacles to introduced the SD-IoV and current yeNs We first presented a generalised SD-IoV architecture, then a layered architecture. Then, we determined the advantages of SD-IoV through three essential processes. In addition, we contrasted the functionality and performance of three different wireless control path implementation methodologies. We outlined the issues with SD-IoV and their current fixes. Finally, we discussed our opinions on the unresolved problems in the SD-IoV field. A car today leaves the factory in the finest possible shape. But in the future, software can be continuously improved while still operating within hardware constraints. This implies that the car may get better even after it leaves the factory.

For instance, by upgrading and improving the features of the car. Software solutions will therefore start to dominate how automakers and fleet managers differentiate themselves in the future. The division of hardware and software is what first makes this paradigm shift possible.

Reference

1. "Software Defined Network based Fog Architecture for Workload Balancing in Autonomous Vehicles," 2022 International Conference on Augmented Intelligence and Sustainable Systems (ICAISS), Trichy, India, pp. 845850, by V. Gowri and B. Baranidharan. doi:10.1109/ICAISS55157.2022.10010 924.
2. S. R. Pokhrel, "Software Defined Internet of Vehicles for Automation and Orchestration," IEEE Transactions on Intelligent Transportation Systems, vol. 22, no. 6, June 2021, pp. 3890–3899, doi: 10.1109/TITS.2021.3077363.
3. S. Aust, "Vehicle Update Management in Software Defined Vehicles," 2022 IEEE 47th Conference on Local Computer Networks (LCN), 261263, Edmonton, Alberta, Canada, 2022, 10.1109/LCN53696.2022.9843360.
4. T. Yuan, W. d. R. Neto, C. E. Rothenberg, K. Obraczka, C. Barakat, and T. Turletti, "Dynamic Controller Assignment in Software Defined Internet of Vehicles Through IEEE Transactions on Network and Service Management, vol. 18, no. 1, March 2021, pp. 585–596, doi: 10.1109/TNSM.2020.3047765, "Multi-Agent Deep Reinforcement Learning."
5. T. -C. Huang, C. -Y. Huang and Y.-C. Chen, "Real-Time D DoS Detection and Alleviation in Software-Defined In-Vehicle Networks," in IEEE Sensors Letters, vol. 6, no. 9, pp. 1-4, Sept. 2022, Artno.6003304,doi:10.1109/LSENS.2022.3202301.
6. Q. Qiu, H. Li, H. Zhang and J. Luo, "Bandit based Dynamic Spectrum Anti-jamming Strategy in Software Defined UAV Swarm Network," 2020 IEEE 11th International Conference on Software Engineering and Service Science (ICSESS), Beijing, China, 2020, pp. 184-188, doi: 10.1109/ICSESS49938.2020.9237733.
7. L. Wan, L. Sun, K. Liu, X. Wang, Q. Lin and T. Zhu, "AutonomousVehicle Source Enumeration Exploiting Non- Cooperative UAV in Software Defined Internet of Vehicles," in IEEE Transactions on Intelligent Transportation Systems, vol. 22, no. 6, pp. 3603-3615, June 2021, doi:10.1109/TITS.2020.3018377.