# An Intent-Based Management Framework for Networks and Security in Software-Defined Vehicles

Jaehoon (Paul) Jeong* and Yiwen Shen*
*Department of Computer Science & Engineering,
Sungkyunkwan University, Suwon, Republic of Korea
Email: {pauljeong, chrisshen}@skku.edu

*Abstract*—**Software-Defined Vehicle (SDV) is a new player towards autonomous vehicles in smart road networks. An SDV is constructed by a software platform like a cloud-native system like Kubernetes and has its internal network. To facilitate the easy and efficient configuration of networks in the SDV, an intent-based management is an appropriate direction. This paper proposes a framework of intent-based management for networks and security in SDVs so that they can communicate with other SDVs and infrastructure nodes for safe driving and infotainment services in the road networks.**

*Index Terms*—**Software-Defined Vehicle, Intent-Based Management, Vehicular Networks, Framework, Interface.**

## I. INTRODUCTION

Software-Defined Vehicle (SDV) is a new paradigm in Intelligent Transportation Systems (ITS). The SDVs interact with each other and infrastructure (e.g., edge servers) for safe driving and infotainment services through vehicular networks, as shown in Fig. 1. SDVs can communicates with each other via Vehicle-to-Vehicle (V2V) communications. They can communicate with infrastructure nodes (e.g., gNodeB in 5G) via Vehicle-to-Infrastructure (V2I) communications.
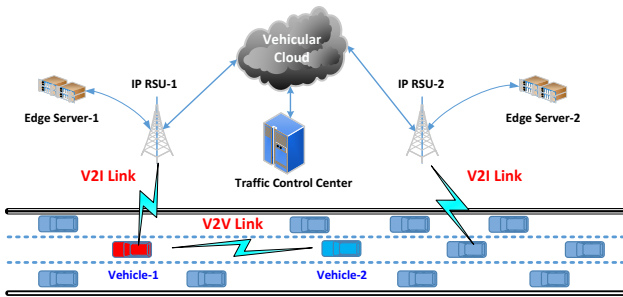


Fig. 1. Vehicular Networks for Software-Defined Vehicles

SDV is regarded as a future platform for autonomous vehicles and electric vehicles. Many automotive-related companies are developing the platform of SDVs, such as Eclipse [1] and AUTOSAR [2]. An SDV consists of computing devices, operating systems, and a cloud-native platform (e.g., Kubernetes [3]). Kubernetes is an open-source platform for managing containerized workloads and services with portability and extendibility. It can facilitate intent-based management with both declarative configuration and automation.

An SDV needs an intent-based management for networks and security inside its in-vehicle networks. An intent is a declarative command to request a configuration for a network or security function [4]. It emphasizes more on "What" is needed (i.e., declarative command) to be accomplished than "How" it should be accomplished (i.e., imperative command). Since there are a huge number of vehicles produced by each automotive company, the networks and security for the SDV needs to be remotely configured and monitored by a control center of each automotive company. The in-vehicle networks are based on Gigabit Ethernet and can be configured as multiple subnets including Electric Control Units (ECU) and infotainment devices. It requires huge overhead for an operator to configure and monitor networks and security for those in-vehicle networks.

This paper proposes a framework of intent-based management for networks and security in SDVs. This framework automates the configuration and monitoring for the networks and security in each SDV through a vehicular cloud and the SDV's mobile network. An SDV User (i.e., administrator) for the management of SDVs can configure and monitor the networks and security through an intent. The intent from the SDV User is delivered to a Cloud Controller in charge of a vehicular cloud for SDVs. The Cloud Controller translates the intent into the corresponding high-level policy, and delivers the high-level policy to an SDV Controller in charge of an SDV. The SDV translates the high-level policy into the corresponding low-level policy and delivered it to an appropriate Network Function (NF) for a specific service (e.g., router, DNS resolver, and firewall) in the SDV.

The remainder of this paper includes the following. Section II proposes an intent-based management framework for SDVs. Section III concludes this paper along with future work.

## II. INTENT-BASED MANAGEMENT FRAMEWORK FOR SOFTWARE-DEFINED VEHICLES

SDVs are managed and monitored by the vehicular cloud. They get help for software updates as well as the configuration of their networks and security from the vehicular cloud. Fig. 1 shows a vehicular network architecture for SDVs. SDVs as vehicles can communicate with each other via V2V and with infrastructure nodes such as IP Road-Side Unit (IP-RSU), for example, gNodeB in 5G networks, respectively. Edge servers can help SDVs to perform their safe driving by processing environmental data collected by the SDVs and giving maneuver guidance to the SDVs.
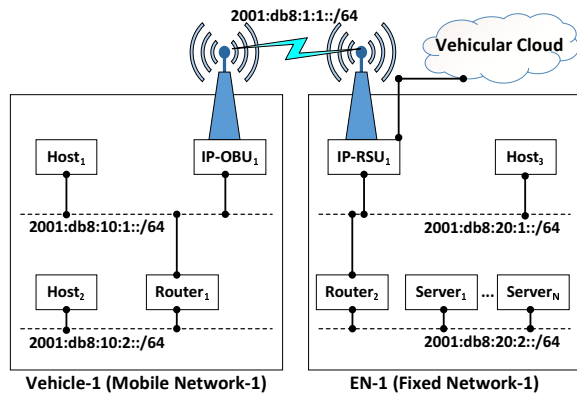
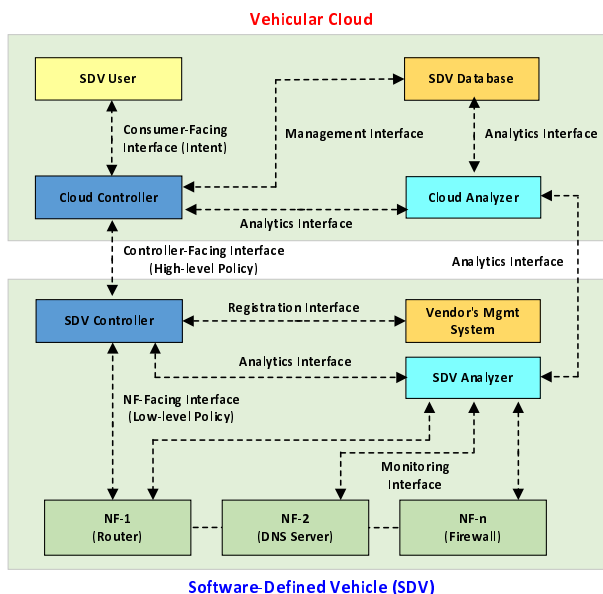Fig. 2. In-Vehicle Network and Edge Network



Fig. 3. Intent-Based Management Framework for Software-Defined Vehicles

An SDV has its own internal networks (called in-vehicle networks), as shown in Fig. 2. The in-vehicle networks consist of multiple subnets connected with each other through routers. IP On-Board Unit (IP-OBU) is a network device in an SDV that has a basic processing ability and can be driven by a low-power CPU (e.g., ARM) with 5G Vehicle-to-Everything (V2X) communication device [5]. IP Road-Side Unit (IP-RSU) is a network device situated along the road as an infrastructure node. It has at least two distinct IP-enabled interfaces where one is for 5G V2X and the other is for the wired network connected to the vehicular cloud [5]. An Edge Network (EN) is a radio access network which has an IP-RSU for wireless communication with other SDVs having an IP-OBU and wired communication with other network devices (e.g., routers, IP-RSUs, and edge servers) [5]. As shown in Fig. 2, the IPv6 prefixes should be configured for the in-vehicle network (called mobile network) and Edge Network (called EN). Also, for V2X IP networking, the wireless interfaces of IP-OBU and IP-RSU should be configured with appropriate IPv6 network prefixes and default gateways towards the infrastructure network connected to the vehicular cloud.

For the automatic network configuration of SDVs, an intent-based management is required between the vehicular cloud and SDVs. Fig. 3 shows a framework of intent-based management for SDVs. The framework consists of a vehicular cloud and SDVs. The vehicular cloud consists of SDV User (as network administrator), Cloud Controller (as an orchestrator for a vehicular cloud), SDV Database (as a main repository for SDV management and monitoring), and Cloud Analyzer (as a monitoring data analyzer for SDVs). The SDV is composed of SDV Controller (as a manager for an SDV), SDV Analyzer (as a monitoring data analyzer for an SDV), Vendor's Management System (as a vendor system to provide cloud-native containers), and Network Functions (NF) such as router, DNS server, and firewall. In this figure, interfaces are defined between a pair of system components in the vehicular cloud and SDV. The intent, high-level policy, and low-level policy can be either XML documents [6] or YAML documents [7]. They can be delivered to the destination components via NETCONF [8], RESTCONF [9], or REST API [10].

## III. CONCLUSION

This paper proposes a framework of intent-based management for networks in a Software-Defined Vehicle (called SDV). Through this intent-based management, SDVs can communicate with other SDVs and infrastructure nodes (e.g., IP-RSU) via V2X communications for safe driving and infotainment services in vehicular networks. As future work, we will design and implement of the system components and interfaces for the intent-based management for SDVs.

## REFERENCES

[1] Eclipse Software Defined Vehicle Working Group Charter. [Online]. Available: https://www.eclipse.org/org/workinggroups/sdv-charter.php

[2] AUTOSAR Adaptive Platform. [Online]. Available: https://www.autosar.org/standards/adaptive-platform

[3] Kubernetes. [Online]. Available: https://kubernetes.io/

[4] 3GPP, "Management and orchestration; Intent driven management services for mobile networks," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 28.312, September 2023, version 18.1.1. [Online]. Available: http://www.3gpp.org/DynaReport/28312.htm

[5] J. P. Jeong, "IPv6 Wireless Access in Vehicular Environments (IPWAVE): Problem Statement and Use Cases," *RFC 9365*, Mar. 2023. [Online]. Available: https://datatracker.ietf.org/doc/rfc9365/

[6] M. Bjorklund, "The YANG 1.1 Data Modeling Language," *RFC 7950*, Aug. 2016. [Online]. Available: https://datatracker.ietf.org/doc/rfc7950/

[7] I. Brian, C. E. Clark, and O. Ben-Kiki, "Yet Another Markup Language (YAML) 1.0," October 2023. [Online]. Available: https://yaml.org/spec/history/2001-05-26.html

[8] R. Enns, M. Bjorklund, J. Schoenwaelder, and A. Bierman, "Network Configuration Protocol (NETCONF)," *RFC 6241*, January 2011. [Online]. Available: https://datatracker.ietf.org/doc/rfc6241/

[9] A. Bierman, M. Bjorklund, and K. Watsen, "RESTCONF Protocol," *RFC 8040*, Jan. 2017. [Online]. Available: https://datatracker.ietf.org/doc/rfc8040/

[10] R. T. Fielding and R. N. Taylor, "Principled Design of the Modern Web Architecture," *ACM Transactions on Internet Technology*, vol. 2, no. 2, May 2002. [Online]. Available: https://doi.org/10.1145/514183.514185