An Intent-Based System for Software-Defined Vehicles in 5G Mobile Networks

Mose Gu*, Yoseop Ahn*, Juwon Hong*, Yiwen Shen†, and Jaehoon (Paul) Jeong*
* Department of Computer Science & Engineering, Sungkyunkwan University, Suwon, Republic of Korea
Email: {rna0415, ahnjs124, hongju2024, pauljeong}@skku.edu
† Department of Software & Computer Engineering, Ajou University, Suwon, Republic of Korea
Email: chrisshen@ajou.ac.kr

***Abstract*—This paper introduces a comprehensive framework for intent-based management of networks, security, and applications in software-defined vehicles (SDVs) within 5G networks. To address the complexities and operational challenges of traditional network management, we propose an Intent-Based system (IBS) that simplifies the administration process by converting high-level policies into actionable commands or configurations. The practical applicability and robustness of this approach are demonstrated through a testbed implemented using the free5GC platform. Our findings confirm the feasibility of intent-based network management and highlight its potential to enhance operational efficiency and scalability in the rapidly evolving automotive industry.**

***Index Terms*—5G network, intent-based networking, intent-based system, network management, automation.**

I. Introduction

The evolution of network management has significantly progressed from manual configurations to sophisticated automated management systems. This progression has fostered the development of Intent-Based Networking (IBN) management and automation [1], [2], driven by increasing complexities, scale, cost efficiency, dynamic environments, service delivery demands, and security requirements [3]. Simultaneously, the automotive industry is undergoing a transformative shift with the emergence of software-defined vehicles (SDVs). These vehicles leverage powerful onboard high-performance computers and a high-speed, often Giga-bit Ethernet-based, network backbone, facilitating the operations of flexible and dynamic functions.

The transition to SDVs introduces a new paradigm in Intelligent Transportation Systems (ITS), enabling direct interactions among vehicles via Vehicle-to-Vehicle (V2V) communications and with interactions with vehicular cloud and edge servers via Vehicle-to-Infrastructure (V2I) communications [4]. Supporting these interactions, the architecture for vehicular networks in SDVs is typically structured into multiple subnets, allowing vehicles to connect with edge servers and the vehicular cloud through IP-based Road-Side Units (IP-RSUs) [5], developing software components that equip vehicles with advanced electronics, sensors, and actuators [5], [6].

The extensive network functions (NF) and application functions (AF) within SDVs necessitate an intent-based management framework to streamline network security and application functions. An intent represents a set of operational goals that the network aims to fulfill, simplifying network management by allowing administrators to focus on outcomes rather than imperative commands or configurations [1], [2]. In response to these demands, this paper seeks to offer a comprehensive and accessible solution for intent-based management of SDVs.

The contributions of this paper are as follows:

- **Framework for Intent-Based Management of SDVs:** This paper proposes a comprehensive framework, along with its components, designed to manage networks, security, and applications in SDVs, simplifying complex operations through intent-based networking.
- **Intent Translator for Policy Conversion:** This paper illustrates an intent translator that efficiently translates an intent into a high-level policy for intent-based SDV management.
- **Testbed Implementation on free5GC:** This paper validates the practicality of our framework with a testbed built on the free5GC platform, demonstrating the feasibility and robustness of intent-based management in real-world scenarios.
- **Automation and Scalability:** This paper provides an accessible and comprehensive approach to intent-based management for SDVs, explaining the design and simplified implementation of the intent translator and its operational efficacy within our testbed.

The remainder of this paper is structured as follows. Section II discusses related work on IBN for SDVs. Section III explains the intent-based management architecture, interactions, and components for SDVs. Section IV details the implementation of the proposed architecture. Section VI concludes our work along with future work.

II. Related Work

Intent-Based Management (IBM) introduces a data-driven paradigm in network and system management, notably within the realm of Software-Defined Vehicles (SDVs). IBM enables the enforcement of user-defined intents within target systems, focusing on achieving specified outcomes and operational goals without dictating the exact steps for execution. This method markedly simplifies network management complexity while boosting scalability and automation. A core element of IBM is the translation of intents, typically expressed in natural language (e.g., English), into high-level policies. This translation is facilitated by Natural Language Processing (NLP) techniques, as demonstrated in prominent studies such as those discussed in Lumi [7]. Deep learning models are employed to interpret and transform these natural language intents into structured high-level policies.

Within the context of SDVs, intent translation into high-level policies is executed by an intent translator for a framework for Interface to Network Security Functions (I2NSF) [8], [9]. The SPT can translate a high-level security policy into the corresponding low-level security policy.

Continuous monitoring and analysis of the activity and performance of a Service Function (SF) such as NF and AF and performance are important to IBM. As highlighted by the I2NSF Working Group [10], ongoing surveillance ensures that SFs work as intended, providing monitoring data for security analytics and enabling the system to identify and rectify any new security attacks. If issues arise, either the high-level or low-level network policies can be revised or new policies can be formulated and implemented to adapt to changing conditions, thus maintaining the I2NSF framework to have responsiveness and adaptability. Several studies have underscored the implementation of IBM in SDVs. For instance, Jeong et al. [11] detail the shift from manual configuration to advanced automatic network management. Their work showcases the critical importance of IBM in managing the complexities and dynamic aspects of vehicular networks. IBM harnesses artificial intelligence (AI), machine learning (ML), software-defined networking (SDN) [12] and Network Functions Virtualization (NFV) [13] to automate network management, ensuring agile service delivery and stringent security policy enforcement a decentralized architecture.

In parallel, the automotive industry is transforming with the advent of SDVs, transitioning from a decentralized architecture to a centralized, architecture with SDN and NFV software-defined architectures that utilize high-performance computing and high-speed networks for flexible and dynamic function provisioning. This shift is reinforced by industry standards and open source projects such as AUTOSAR [14], Eclipse SDN [6] and SOAFEE [15], integrating SDVs with a central cloud and an edge cloud. The Connected Vehicle Systems Alliance (COVESA) [16] has further strengthen this transformation by developing the Vehicle Signal Specification (VSS) for vehicle data models for IBM in SDVs. This integration effectively supports advanced driver-assistance systems (ADAS) and autonomous driving through reliable, fast data exchange via 5G vehicle to Everything (5G V2X) communications [4].

Collectively, the integration of IBM in SDVs marks a significant advancement in the domain of vehicular networks highlighting its potential to enhance operational efficiency, scalability, and automation, and also providing a comprehensive framework for managing the intricate and dynamic environments in road networks.

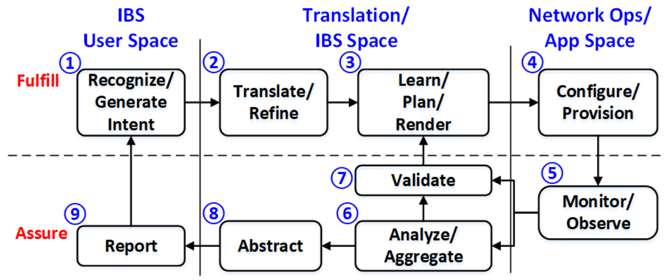

Fig. 1. The Life Cycle of Intent-Based System for SDV Management

III. Intent-Based Management Framework for Software-Defined Vehicles

An Intent-Based Management Framework is a software framework to provide SDVs with intelligent user services, such as ADAS and autonomous driving. This section explains the intent-based management framework specifically designed for SDVs [17]. It begins by outlining the life cycle of an IBS for managing SDVs with intents. Then, it details the essential components and interfaces that constitute the framework.

A. *Intent-Based Management Framework for SDVs*

This subsection describes the key component of the framework, IBM framework for SDVs. To enable automatic network configuration of Software-Defined Vehicles (SDVs), an intent-based management system is necessary to facilitate interaction between the vehicular cloud and SDVs. Fig 2 illustrates a framework for intent-based management in SDVs, comprising both the vehicular cloud and the SDVs themselves [18]. The vehicular cloud is composed of several critical components: the SDV User interface (serving as the network administrator), the Cloud Controller (acting as the orchestrator for the vehicular cloud), the SDV Database (which functions as the primary repository for managing and monitoring SDVs), and the Cloud Analyzer (which processes and evaluates monitoring data for SDVs, similar to the Network Data Analytics Function (NWDAF) in 5G networks [19], [20].

- **SDV User:** It is typically accessed via a web-browser interface, allows SDV administrators to input network intents for the SDV controllers.
- **Cloud Controller:** It manages and coordinates the various system elements within the vehicular cloud. It translates the security service intents from the SDV User into corresponding security policies and selects the appropriate service functions (SFs) to implement these policies.
- **Cloud Vendor's Management System:** It provides virtualized service function images for the vehicular cloud services and manages their registration and access details with the Cloud Controller.
- **Cloud Analyzer:** It collects and analyzes monitoring data from SDV Analyzers to verify the performance and functionality of service functions, akin to the Network Data Analytics Function (NWDAF) in 5G networks.

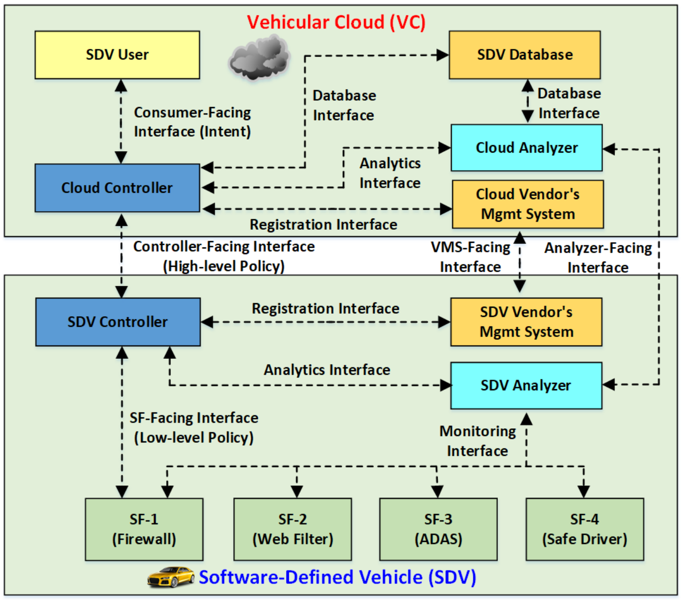

Fig. 2. An Intent-Based Management Framework for SDVs

- **SDV Database:** It is a comprehensive repository for managing SDV configurations, including network settings, security information, and navigation data.

The Intent-Based System (IBS) within an SDV includes components such as the SDV Controller, SDV Analyzer, SDV Vendor's Management System, and various Service Functions (SFs) like routers, DNS servers, firewalls, and applications (e.g., safe driving assistants and navigators) which are described as follows.

- **SDV Controller:** It oversees and manages the internal components of the SDV framework. It converts high-level policies received from the Cloud Controller into low-level policies that service functions can execute.
- **SDV Vendor's Management System:** It provides virtualized service function images for SDVs and registers their capabilities and access information with the SDV Controller.
- **Service Function (SF):** It refers to components that can be virtual network functions (VNFs), cloud-native network functions (CNFs), or physical network functions (PNFs). These functions deliver various services, including security (firewalls, DDoS mitigation) and network/application services.
- **SDV Analyzer:** It collects and analyzes monitoring data from the service functions within SDVs to ensure their performance and activity. It functions similarly to the NWDAF in 5G networks, identifying issues such as security threats, traffic congestion, or performance degradation and providing feedback or policy reconfigurations to the SDV Controller for troubleshooting and management.

B. Interfaces in the SDV Management Framework

This subsection describes the interfaces shown in Fig 2. The framework includes several interfaces for communication between the vehicular cloud components and SDVs which are described as follows.

- **Consumer-Facing Interface:** It Connects the SDV User to the Cloud Controller for intent delivery.
- **Controller-Facing Interface:** It Facilitates the transmission of high-level policies from the Cloud Controller to the SDV Controller.
- **SF-Facing Interface:** It Enables the delivery of low-level policies from the SDV Controller to the service functions.
- **Registration Interface:** It Transfers service function capabilities and access information for registration between the Cloud Controller and the Cloud Vendor's Management System or between the SDV Controller and the SDV Vendor's Management System.
- **Monitoring Interface:** It is Used by the SDV Analyzer to collect monitoring data from service functions, identifying and resolving security, system, and network issues.
- **Analytics Interface:** It Delivers policy reconfigurations or feedback based on analyzed monitoring data. This interface connects the SDV Analyzer to the SDV Controller, Cloud Analyzer, or Cloud Controller.
- **Analyzer-Facing Interface:** It facilitates the exchange of analysis data between the SDV Analyzer and the Cloud Analyzer.
- **VMS-Facing Interface:** It Manages the exchange of service function container images and feature information between the Cloud Vendor's Management System and the SDV Vendor's Management System.
- **Database Interface:** It Allows data exchange within the SDV database, connecting it to the Cloud Controller or Cloud Analyzer.

Intent, high-level policies, and low-level policies can be formatted as XML [21], [22] or YAML documents [23] and transmitted using NETCONF [24], RESTCONF [25], or REST APIs [26].

C. Life Cycle of an Intent-Based System for Intelligent SDV Management

Based on the IBN principles described in RFC 9315 [2], the life cycle of an Intent-Based System (IBS) for intelligent SDV management is divided into three distinct domains: the SDV User Space, the Translation and IBS Space, and the Network Operations (Ops) and Application Space. Each domain is further segmented into functions for fulfillment and assurance. The fulfillment function covers steps from the initial intent input to the final configuration and the provisioning of network functions (NFs) and application functions (AFs) within an SDV. Conversely, the assurance function focuses on analyzing monitoring data to analyze the functionality and performance of these functions and applications.

Fig. 1 illustrates the life cycle of IBN for managing SDVs. This life cycle is divided into three primary spaces: IBS User Space, Translation and IBS Space, and Network Operations (Ops) and Application (App) Space. Each of these spaces is further divided into two functions: fulfillment and assurance.

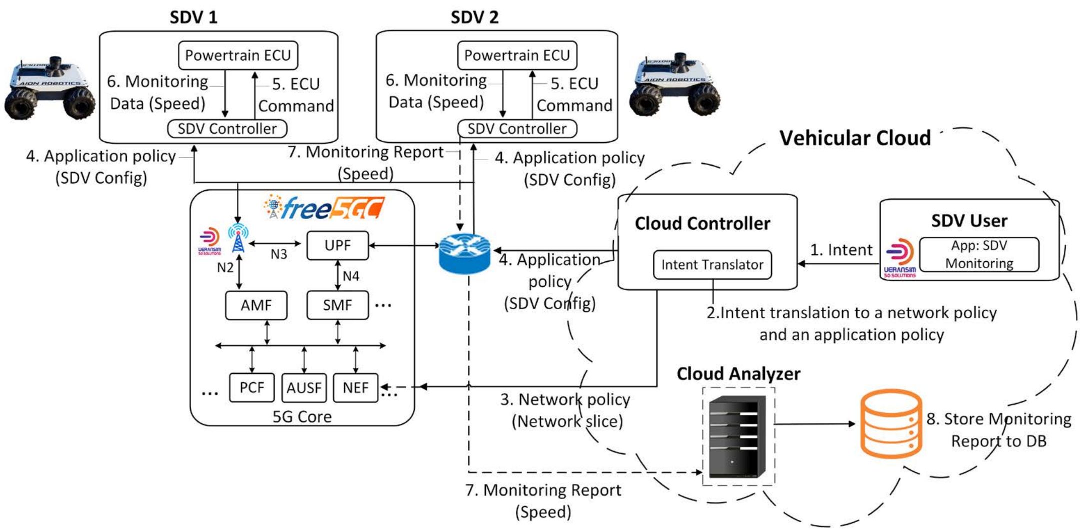

Fig. 3. A Testbed for Intent-Based System for SDVs

In the IBS User Space, the IBS User (i.e., SDV User) defines high-level intents, which articulate the desired operational goals and outcomes for the SDVs, which are expressed in natural language (e.g., English). These intents are submitted to the Cloud Controller having an intent translator via the Consumer-Facing Interface.

In the Translation and IBS Space, the Cloud Controller plays a central role by receiving the high-level intents from the IBS User and translating them into high-level policies through its Intent Translator. This process ensures that the intents are converted into precise and actionable policies. The high-level policies are then delivered to the SDV Controllers via the Controller-Facing Interface.

The Cloud Vendor's Management System interacts with the Cloud Controller through the Registration Interface, providing SF images and relevant configuration data. The Cloud Analyzer processes monitoring data from the SDVs and communicates with the Cloud Controller through the Analytics Interface for closed-loop control. Additionally, the SDV Database stores management and operational data, supporting data-driven decisions and historical analysis for SDVs.

In the Network Operations and Application Space, the SDV Controller receives high-level policies from the Cloud Controller. The SDV controller translates the high-level policies into the low-level poliocies. It then delivers them to appropriate its SFs for the intended services in an SDV. These SFs, including firewall, anti-virus, safe driving application, and navigators, execute the low-level policies to achieve the desired operational states. The SDV Vendor's Management System provides SF images and configuration data to the SDV Controller via the Registration Interface. The SDV Analyzer continuously monitors the functionality, performance, and activities of the SFs, ensuring compliance with the high-level policies. It sends monitoring reports back to the Cloud Analyzer through the Analyzer-Facing Interface, enabling real-time adjustments and optimizations.

This structured life cycle framework ensures that SDVs operate efficiently and effectively with real-time monitoring and dynamic policy adjustments with the vehicular cloud. Thus, the intent-based management system, enables scalable and automated network configurations, thereby enhancing the overall functionality and performance of SDVs in the ITS.

D. A Use Case of Intent-Based Management Framework for SDVs

A use case of an intent-based system (IBS) for SDVs within a 5G network and experiment of an IBS for SDVs within a 5G network using the FREE 5GC. This consists of multiple key components, including SFs, the SDV Controller, the Cloud Controller, and the Cloud Analyzer. They interact with each other within the vehicular cloud and SDV. As shown in Figs 3, 4 The following steps detail how components interact within the framework:

- **Steps 1 and 2:** The SDVs (e.g., SDV 1 and SDV 2) contain a Powertrain Electronic Control Unit (ECU) that monitor operational data, such as speed and direction, and executes commands for such monitoring issued by the SDV Controller. The operational flow begins with the SDV User (e.g., administrator), who utilizes an application (SDV Monitoring) to send an intent to the Cloud Controller. This intent is translated into high-level policies (i.e., a network policy and application policy) by the Intent Translator within the Cloud Controller.
- **Step 3:** The translated network policy, which may include a network slicing request, is delivered to the 5G core network components, including the Access and Mobility Management Function (AMF), Session Management Function (SMF), User Plane Function (UPF), Policy Control Function (PCF), Authentication Server Function (AUSF), and Network Exposure Function (NEF).
- **Step 4:** Concurrently, the application policy detailing SDV configuration is delivered to the respective SDV Controllers.

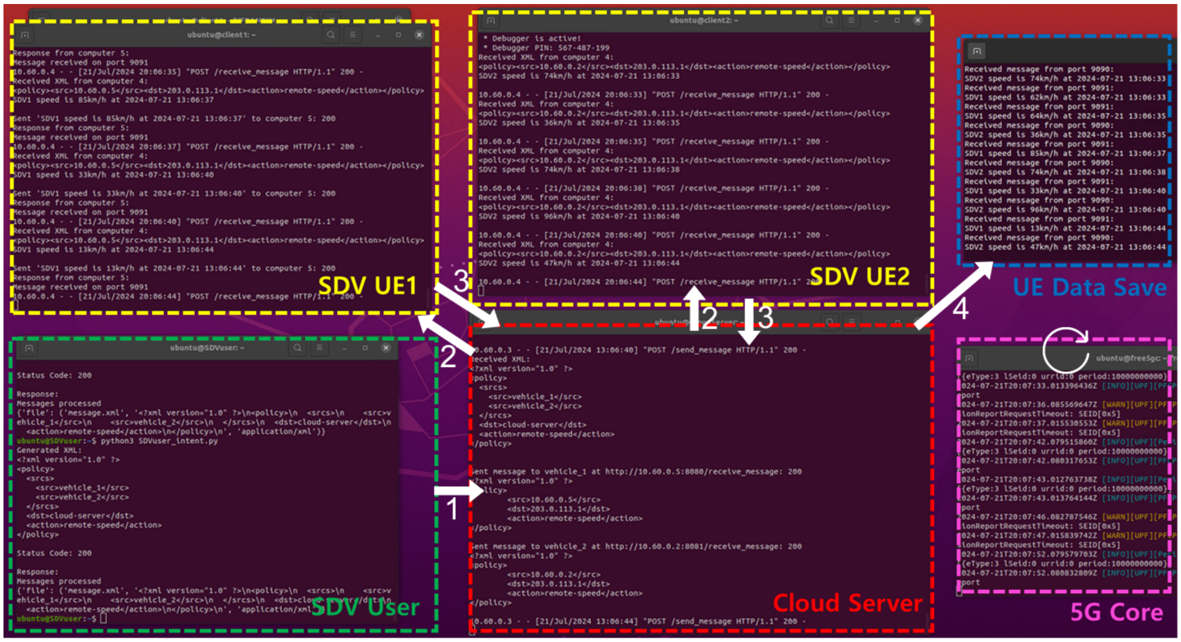

Fig. 4. A Demonstration for Intent-Based System for SDVs

- **Step 5:** Each SDV Controller applies the received application policy to the Powertrain ECU, adjusting the vehicle's operational parameters accordingly.
- **Step 6:** The SDVs continuously monitor their operational data, such as speed, direction, and send this monitoring data back to the SDV Controllers.
- **Step 7:** The SDV Controllers compile this data into monitoring reports, which are then sent to the Cloud Analyzer.
- **Step 8:** The Cloud Analyzer processes the monitoring reports to evaluate the functionality and performance of the applied policies. It stores the monitoring data in the database for future reference and analysis.

This closed-loop intent-based management system supports scalable and automated network management, thereby enhancing the overall functionality and performance of intelligent transportation systems (ITS).

IV. Implementation of IBS for SDVs

The architecture consists of several components such as: two SDV User Equipment (UEs), such as SDV 1 and SDV 2, an SDV User APP, and a vehicular cloud server that performs the intent translation. These components interact with each other within an emulated network empowered by the Free 5GC platform which is a fully functional and modular 5G core network implementation.

The testbed was implemented on virtual machines, which were configured with Ubuntu 20.04 LTS, chosen for its stability and extensive support. Each component of the SDV framework-the SDVs, SDV User APP, and Cloud Server—runs in isolated virtual environment to mimic a real-world deployment scenario and ensure reliable inter-component communication. The key of our network setup is the Free 5GC platform, which serves as the 5G core. This platform is set up to manage network functions (NFs) such as AMF, SMF, UPF, PCF and NEF. This configuration allows for the precise emulation of 5G network behaviors (e.g., network slicing and policy enforcement).

Shown in Fig. 4, the functionality of our implemented testbed was demonstrated where our vehicle was continuously monitored emulated SDVs. Note that SDV 1 and SDV 2 operate as distinct entities, sending and receiving commands and data, emulating actual vehicle interactions using UERANSIM. The flow of the testbed is as follows:

1) The SDV User sends high-level intents to the Cloud Server. These intents describe what the SDV should do (e.g., control vehicle speed) without specifying how to achieve or execute the intents.
2) The Cloud Server receives these intents and translates them into XML-based high-level policies. The intent translation is done by mapping the high-level intents to high-level network or application policies. After the translation, The Cloud Server sends the high-level policies to the targeted SDVs.
3) Upon receiving such policies, each SDV controller in SDV UE1 and SDV UE2 performs the policy translation. In this translation, the SDV Controller translates the high-level policy to an actionable low-level policy. After translating, the vehicles configure themselves to prepare for reporting their speeds to the Cloud Server.
4) SDV UE1 and SDV UE2 send their monitoring data including their speed to the database on the Cloud Server. The UE Data Save Component shows that the vehicle's data such as current speed at various timestamps is stored in the database periodically.

This testbed demonstrates the practical applicability of our IBS for SDVs within the 5G networks, highlighting the advancements in network management and operational efficiency. Note that for simplicity, the intent translation was performed by the fixed mapping form an intent to an application policy. In this paper, the translation of a network policy was not performed. The automatic intent translation with both application and network policies are left as future work.

V. Research Challenges for IBS for SDVs

We have the following research challenges for the Intent-Based System (called IBS) for SDVs.

1) An Intent Translator: The design and implementation of an intent translator is required for correct and efficient intent translation into both a network policy and an application policy.
2) A Closed-Loop Intent Control: A closed-loop control system should be designed and implemented so that an intent can be enforced into the SFs in SDVs and the translated network and application policies can work correctly and optimally in the SFs,
3) On-line Diagnosis of SFs in SDVs: We need to make data analytics models (e.g., Machine Learning and Deep Learning models) for on-line diagnosis of SFs in SDVs for safety and security.

VI. Conclusion

This paper presents a comprehensive framework for intent-based management of networks, security, and applications in software-defined vehicles (SDVs) using 5G networks. The framework's key feature is an intent translator that simplifies network management by converting high-level intents into executable policies. While the current testbed implementation is limited to predefined mappings for application policies, it still demonstrates the practical potential of this intent-based system (IBS). The framework shows promise in improving operational efficiency and reducing operational cost, particularly as the number of SDVs increases. As future work, we will focus on developing automatic intent translation for both application and network policies, and integrating advanced machine learning techniques to further optimize network functions and security protocols.

Acknowledgments

This work was supported by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korean government (MSIT)(No. RS-2024-00398199 and No. 2022-0-01015, Development of Candidate Element Technology for Intelligent 6G Mobile Core Network). Note that Jaehoon (Paul) Jeong is the corresponding author.

References

[1] "Intent Driven Management Services for Mobile Networks," September 2022. [Online]. Available: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3554

[2] A. Clemm, L. Ciavaglia, L. Z. Granville, and J. Tantsura, "Intent-Based Networking - Concepts and Definitions," *RFC 8192*, October 2022. [Online]. Available: https://www.rfc-editor.org/info/rfc9315

[3] A. Leivadeas and M. Falkner, "A Survey on Intent-Based Networking," *IEEE Communications Surveys & Tutorials*, vol. 25, no. 1, pp. 625–655, 2023.

[4] 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/

[5] AUTOSAR, Adaptive Platform Working Groups. AUTOSAR Adaptive Platform. [Online]. Available: https://www.autosar.org/standards/adaptive-platform

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

[7] A. Jacobs, R. Pfitscher, R. Ribeiro, R. Ferreira, L. Granville, W. Willinger, and S. Rao, "Hey, Lumi! Using Natural Language for Intent-Based Network Management," in *USENIX Annual Technical Conference 2021*, July 2021. [Online]. Available: https://www.usenix.org/conference/atc21/presentation/jacobs

[8] P. Lingga, J. Jeong, J. Yang, and J. Kim, "SPT: Security Policy Translator for Network Security Functions in Cloud-Based Security Services," *IEEE Transactions on Dependable and Secure Computing*, pp. 1–14, 2024.

[9] D. Lopez, E. Lopez, L. Dunbar, J. Strassner, and R. Kumar, "Framework for Interface to Network Security Functions," *RFC 8329*, Feb. 2018. [Online]. Available: https://datatracker.ietf.org/doc/rfc8329/

[10] "I2NSF Working Group," Oct. 2023. [Online]. Available: https://datatracker.ietf.org/wg/i2nsf/about/

[11] Y. Shen, Y. Ahn, M. Gu, and J. P. Jeong, "Intent-based management for software-defined vehicles in intelligent transportation systems," in *2024 IEEE 10th International Conference on Network Softwarization (NetSoft)*, 2024, pp. 1–6.

[12] R. Marin-Lopez, G. Lopez-Millan, and F. Pereniguez-Garcia, "A YANG Data Model for IPsec Flow Protection Based on Software-Defined Networking (SDN)," *RFC 9061*, Jul. 2021. [Online]. Available: https://datatracker.ietf.org/doc/rfc9061/

[13] "Network Functions Virtualisation (NFV); Architectural Framework," December 2014. [Online]. Available: https://www.etsi.org/deliver/etsi_gs/nfv/001_099/002/01.02.01_60/gs_nfv002v010201p.pdf

[14] AUTomotive Open System ARchitecture (AUTOSAR) Adaptive Platform. [Online]. Available: https://www.autosar.org/standards/adaptive-platform

[15] Scalable Open Architecture for Embedded Edge (SOAFEE). [Online]. Available: https://www.soafee.io/

[16] The Connected Vehicle Systems Alliance (COVESA). [Online]. Available: https://covesa.global/

[17] J. P. Jeong, Y. C. Shen, Y. Ahn, and M. Gu, "An Intent-Based Management Framework for Software-Defined Vehicles in Intelligent Transportation Systems," *draft-jeong-opsawg-intent-based-sdv-framework-02*, Jun. 2024. [Online]. Available: https://datatracker.ietf.org/doc/draft-jeong-opsawg-intent-based-sdv-framework/

[18] J. P. Jeong, Y. Ahn, Y. Kim, and J.-S. Park, "Intent-Based Network Management Automation in 5G Networks," *draft-jeong-nmrg-ibn-network-management-automation-03*, Nov. 2023. [Online]. Available: https://datatracker.ietf.org/doc/draft-jeong-nmrg-ibn-network-management-automation/

[19] "Architecture Enhancements for 5G System (5GS) to Support Network Data Analytics Services," September 2022. [Online]. Available: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3579

[20] "Network Data Analytics Services," September 2022. [Online]. Available: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3355

[21] B. Martin, "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)," *RFC 6020*, october 2010. [Online]. Available: https://www.rfc-editor.org/info/rfc6020

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

[23] 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

[24] 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/

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

[26] 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