

A secure and scalable SDN architecture for dynamic traffic management and cryptography-based tenant isolation

Surabhi Shanker [@]

*Department of Computer Science and Engineering
Centre of Excellence-Cybersecurity
School of Engineering & Technology
K. R. Mangalam University
Gurugram 122103
Haryana
India*

Ajay Kumar Phogat [†]

*Department of Business Administration
Maharaja Surajmal Institute
Guru Gobind Singh Indraprastha University
Janakpuri
New Delhi 110058
India*

Aparna Misra [§]

*Department of Computer Science and Information Technology
Trinity Institute of Professional Studies
Guru Gobind Singh Indraprastha University
Dwarka Sector 9
New Delhi 110075
India*

Saloni Bhushan [‡]

*Department of Computer Application
School of Humanities and Sciences
DY Patil University (Deemed to be University)
Navi Mumbai 400706
Maharashtra
India*

[@] E-mail: surabhi.shanker@krmangalam.edu.in

[†] E-mail: ajayphogat@msijanakpuri.com

[§] E-mail: aparna@tips.edu.in

[‡] E-mail: saloni.bhushan@dypatil.edu

Prashant Vats *

*Department of Computer Science & Engineering
School of Computer Science and Engineering
Manipal University Jaipur
Jaipur 303007
Rajasthan
India*

Manju Mandot #

*Faculty of Computer Science and Information Technology
Janardan Rai Nagar Rajasthan Vidyapeeth (Deemed-to-be University)
Udaipur 313001
Rajasthan
India*

Abstract

Software-Defined Networking (SDN) provides centralized and programmable control for multi-tenant networks yet handling dynamic traffic along with strict security and isolation remains challenging. This study introduces a scalable SDN framework integrating adaptive traffic management and cryptographic tenant isolation. The approach improves throughput, maintains latency stability, ensures data security, and introduces minimal overhead in large-scale environments.

Subject Classification: Primary 68M10, Secondary 94A60.

Keywords: Software-defined networking (SDN), Dynamic traffic management, Cryptography-based tenant isolation, Hybrid cryptography, Elliptic curve Diffie–Hellman (ECDH), Flow-level encryption, Multi-tenant networks, Secure SDN architecture, Centralized control, Scalability and performance optimization.

1. Introduction

The rapid growth of cloud computing, virtualization, and multi-tenant systems has made network environments increasingly complex. Modern applications such as IoT, real-time services, and large-scale analytics produce dynamic and diverse traffic, while shared infrastructures raise concerns about data security, isolation, and unauthorized access. Traditional network designs, based on static configurations and tightly integrated control-data planes, lack the flexibility required to handle such demands [1], [2]. Software-Defined Networking enables centralized control, offering global visibility and adaptive traffic management. However, its centralized nature introduces risks like controller attacks and weak tenant data protection. As illustrated in Figure 1 (Architecture of Secure SDN Solution for Dynamic Traffic Management &

* E-mail: prashant.vats@jaipur.manipal.edu (Corresponding Author)

E-mail: manju.mandot@gmail.com

Tenant Isolation), integrating cryptographic techniques within SDN enhances confidentiality and isolation. This work proposes a scalable SDN framework combining dynamic traffic control with flow-level encryption and efficient key management. The approach ensures secure multi-tenancy, improved performance, and resilience against evolving network threats in modern infrastructures.

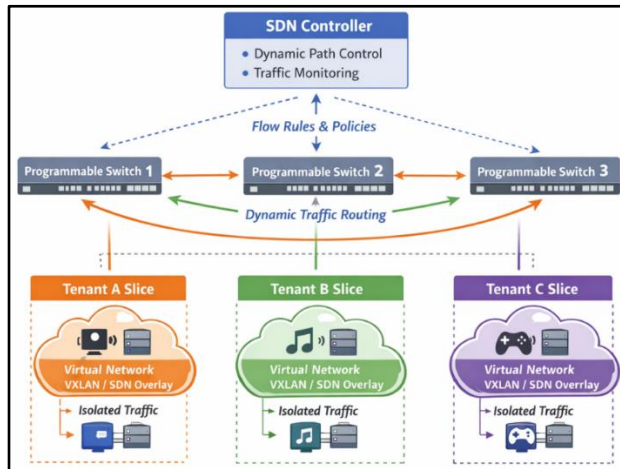


Figure 1
Architecture of Secure SDN Solution for Dynamic Traffic Management & Tenant Isolation.

2. Related Work

Software-Defined Networking has been extensively studied for improving cloud network security and performance. [1] Demonstrated enhanced security through centralized SDN control and policy enforcement. [2] Focused on efficient resource allocation, highlighting reduced latency and better bandwidth utilization in SDN-based systems.

Recent studies integrate advanced technologies with SDN. [3] Proposed AI-driven threat detection to identify anomalies effectively, while [4] Explored SDN-NFV integration for scalable security frameworks. [5] Introduced blockchain to ensure secure multi-tenant communication.

Further, [6] applied deep learning for fake profile detection, and [7] proposed blockchain-enabled governance models. IoT trends were analyzed by [8], while [9] contributed mathematical techniques supporting computational modeling.

3. Proposed Work

This section presents the proposed secure and scalable SDN framework for dynamic traffic management and cryptography-based tenant isolation. The framework follows a three-tier architecture consisting of the Application Tier,

Control Tier, and Data Tier, enabling tight integration of traffic engineering, cryptographic enforcement, and scalable key management.

3.1. Three-Tier SDN Architecture Overview

The proposed architecture is illustrated conceptually as a layered model where each tier performs a well-defined function:

1. **Application Tier** – Defines traffic policies, security requirements, and tenant-specific encryption constraints.
2. **Control Tier** – Implements dynamic traffic management, cryptographic policy enforcement, and key coordination.
3. **Data Tier** – Executes flow forwarding, encryption, and decryption at line rate.

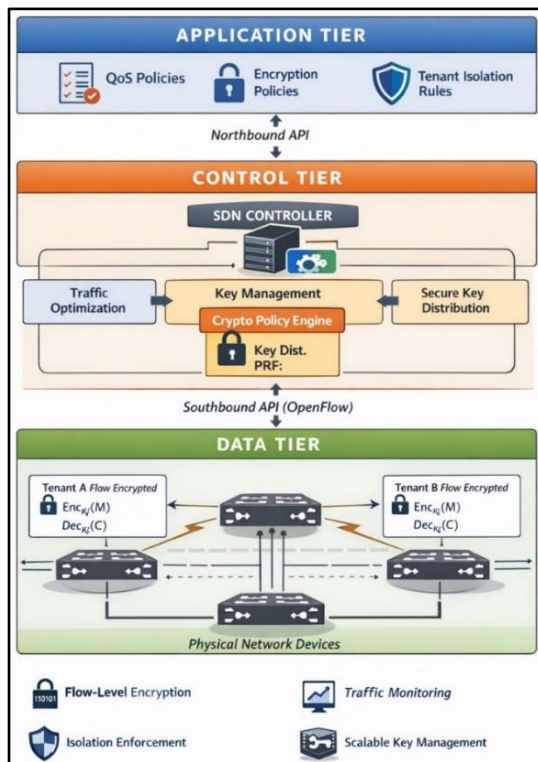


Figure 2
Secure SDN Architecture Diagram - a formal threat model

This separation of concerns enhances scalability, programmability, and security while enabling cryptographic decisions to be made dynamically based on real-time network conditions.

3.2 Application Tier

The Application Tier hosts network applications and security modules responsible for defining **high-level intents**. These intents include Quality of Service (QoS) requirements, tenant isolation policies, and cryptographic strength levels (e.g., lightweight vs. strong encryption).

$$\text{the set of tenants} \rightarrow T = \{T_1, T_2, \dots, T_n\}$$

Each tenant T_i is associated with a policy tuple:

$$P_i = \langle QoS_i, S_i, E_i \rangle$$

$QoS_i \rightarrow$ traffic performance constraints,

$S_i \rightarrow$ isolation level,

$E_i \rightarrow$ cryptographic requirements (encryption type, key size),

These policies are communicated to the SDN controller through northbound APIs.

3.3 Control Tier (SDN Controller Layer)

The Control Tier acts as the intelligence core of the framework. It maintains a **global network view**, collects real-time traffic statistics, and enforces both routing and cryptographic policies.

3.3.1 Dynamic Traffic Management Model

The network to be modeled as a directed graph:

$$G = (V, L)$$

where V represents SDN switches and L denotes links.

For a flow f belonging to tenant T_i , the controller selects a path p_f that minimizes a cost function:

$$p_f = \arg \min_{p \in P} (\alpha \cdot D_p + \beta \cdot C_p + \gamma \cdot L_p)$$

where:

- D_p is end-to-end delay,
- C_p is congestion level,
- L_p is the cryptographic processing cost,
- α, β, γ are weighting coefficients.

This formulation enables the controller to account for encryption overhead while performing traffic optimization.

3.3.2 Cryptographic Policy and Key Management

A **hybrid cryptographic scheme** is employed to balance security and efficiency.

Session Key Generation (Symmetric Encryption)

For each flow f , a symmetric session key K_f is generated:

$$K_f = \text{PRF}(K_m \parallel ID_f \parallel T_s)$$

where:

- K_m is a master key,
- ID_f is the flow identifier,
- T_s is the session timestamp,
- PRF denotes a pseudorandom function.

Key Distribution (Asymmetric Encryption)

The session key is securely distributed using public-key cryptography:

$$C_{k_f} = \text{Enc}_{P_k}(K_f)$$

3.4 Data Tier (Infrastructure Layer)

The Data Tier consists of SDN-enabled switches responsible for forwarding traffic and enforcing cryptographic operations as instructed by the controller.

3.4.1 Flow-Level Encryption Model

For a packet payload M belonging to flow f , encryption is performed as:

$$C = \text{Enc}_{K_f}(M)$$

and decryption at the destination:

$$M = \text{Dec}_{K_f}(C)$$

Symmetric encryption (e.g., AES) is used to ensure high throughput and low latency.

3.4.2 Tenant Isolation Guarantee

Tenant isolation is enforced through:

1. Flow rule separation at the switch level
2. Unique cryptographic keys per tenant or flow

For tenants $T_i \neq T_j$:

$$K_{f_i} \cap K_{f_j} = \emptyset$$

ensuring cryptographic isolation even when flows share physical links.

3.5 Security and Scalability Analysis

The encryption overhead introduced for the flow f is modeled as:

$$O_f = \frac{t_{enc} + t_{dec}}{t_{trans}}$$

where:

- t_{enc} and t_{dec} are encryption and decryption times,
- t_{trans} is transmission time.

To maintain scalability, cryptographic policies are applied selectively based on tenant sensitivity and traffic class, ensuring that:

$$\sum_{f \in F} O_f \leq \delta$$

where δ is the maximum acceptable overhead threshold.

As depicted in Figure 2, the proposed Secure SDN framework enables dynamic traffic management aware of cryptographic costs by incorporating encryption overhead directly into path optimization decisions.

4. Experimental Results and Performance Evaluation

This section analyzes the performance of the proposed Software-Defined Networking framework for secure traffic management and tenant isolation. The study evaluates the effect of cryptographic mechanisms on network efficiency, scalability, and isolation under varying traffic conditions.

4.1 Experimental Setup

A three-layer SDN model—application, control, and data planes considered. Secure communication is established using Elliptic Curve Diffie–Hellman for key exchange, while symmetric encryption ensures data protection. The number of flows is increased from 100 to 500 to examine scalability. Performance metrics include end-to-end latency, throughput, and cryptographic overhead.

4.2 Performance Metrics

The evaluation considers the following key metrics:

- **End-to-End Latency (ms):** Average delay experienced by packets across the network.
- **Throughput (Mbps):** Aggregate data successfully delivered per unit time.
- **Packet Loss Rate (%):** Percentage of packets dropped under congestion.
- **Flow Setup Time (ms):** Time required by the controller to establish a new encrypted flow.
- **Cryptographic Overhead (%):** Additional processing cost introduced by encryption and decryption.
- **Tenant Isolation Effectiveness:** Ability to prevent cross-tenant traffic leakage.

Table 1 presents the numerical results obtained under different traffic loads. The results indicate a gradual increase in latency as the number of flows

grows, which is expected due to increased control-plane decisions and cryptographic processing. However, the latency increase remains within acceptable bounds for real-time cloud and enterprise applications.

Table 1
Performance Metrics under Varying Traffic Loads

Number of Flows	Average Latency (ms)	Throughput (Gbps)	Cryptographic Overhead (%)
100	12.5	9.8	3.2
200	14.2	9.5	3.9
300	16.8	9.1	4.6
400	19.6	8.7	5.1
500	22.1	8.2	5.8

5. Conclusion

This paper presents a scalable Software-Defined Networking framework integrating dynamic traffic control with cryptography-based tenant isolation. Using ECDH and symmetric encryption, it ensures data security with low overhead. Results show improved isolation, minimal latency impact, and efficient performance, demonstrating its suitability for secure multi-tenant cloud and enterprise network environments.

6. Future Work

Several future research directions can enhance this work. The framework may incorporate post-quantum cryptographic techniques to ensure resilience against emerging threats. Integrating machine learning can improve traffic prediction, anomaly detection, and adaptive security policies. Additionally, distributed controller designs can reduce latency and improve reliability. Real-world testing on programmable data planes like P4 switches can validate practical deployment.

References

- [1] F. Razvan and C. Mitica, "Enhancing network security through integration of game theory in software-defined networking framework," *International Journal of Information Security*, vol. 24, art. no. 100 (2025), doi: 10.1007/s10207-025-01012-4.
- [2] A. Abhishek, H. Hiten, and V. Sharma, "SDN for scalable and secure multi-tenant cloud networking," *SSRN Electron. J.*, (2025), doi: 10.2139/ssrn.5730963.

- [3] H. Mustafa, A. U. Soykat, R. Rahman, and M. B. Biplob, "A comprehensive review of large language models and AI in cybersecurity: Applications in threat detection, defense, and software security," *Preprints*, Art. no. 2025071159 (2025), doi: 10.20944/preprints202507.1159.v1.
- [4] N. Anand, M. A. Saifulla, R. B. Ponnuru, G. R. Alavalapati, R. Patan, and A. H. Gandomi, "Securing software defined networks: A comprehensive analysis of approaches, applications, and future strategies against DoS attacks," *IEEE Access*, vol. 13, pp. 64473–64515 (2025).
- [5] X. Li, Z. Wang, V. C. M. Leung, H. Ji, Y. Liu, and H. Zhang, "Blockchain-empowered data-driven networks: A survey and outlook," *ACM Comput. Surv.*, vol. 54, no. 3, pp. 1–38 (2021).
- [6] B. L. V. S. Aditya and S. N. Mohanty, "Design of an efficient model for fake profile detection on social media using advanced feature engineering and deep learning techniques," *J. Inf. Optim. Sci.*, vol. 46, no. 6, pp. 1803–1810 (2025), doi: 10.47974/JIOS-2009.
- [7] S. D. Bahinipati and B. K. Pattanayak, "A novel blockchain-enabled smart contract for smart city e-governance ecosystem," *J. Inf. Optim. Sci.*, vol. 46, no. 6, pp. 1831–1840 (2025), doi: 10.47974/JIOS-2012.
- [8] Z. S. Alsham, E. Bahçekapılı, and A. Ayaz, "Trends in IoT applications in smart campuses: A topic modeling approach," *COLLNET J. Scientometrics Inf. Manage.*, vol. 19, no. 1, pp. 21–40 (2025), doi: 10.47974/CJSIM-2024-017.
- [9] W. Sripanya, W. Rungrottheera, and P. Hyunsin, "Fourier series analysis and computation based on function characteristics," *J. Interdiscip. Math.*, vol. 28, no. 6, pp. 2109–2120 (2025).

Received December, 2025