iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** V **Month of publication:** May 2026

DOI: https://doi.org/10.22214/ijraset.2026.82544

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue V May 2026- Available at www.ijraset.com

Deep Packet Inspection Engine for Real-Time Network Traffic Classification and Rule-Based Filtering

Priyanshu Pandey[1], Sanskar Main[2]
Department of Artificial Intelligence, Madhav Institute Of Technology, Gwalior

Abstract: ***This paper presents a Deep Packet Inspection (DPI) engine capable of packet parsing, TLS Server Name Indication (SNI) extraction, traffic classification, and rule-based filtering. The proposed system supports both single-threaded and multi-threaded architectures for scalable network monitoring and cybersecurity analysis. The engine processes PCAP traffic, extracts flow-level metadata using five-tuple identification, classifies encrypted traffic using TLS SNI fields, and applies application/domain-based filtering policies. Experimental evaluation demonstrates successful traffic classification and efficient packet processing suitable for educational and enterprise-level cybersecurity applications.***

Keywords: Deep Packet Inspection, Cybersecurity, TLS, SNI Extraction, Network Security, Traffic Classification, Multi-threading.

I. INTRODUCTION

Modern internet traffic is heavily encrypted using TLS and HTTPS protocols, making traditional packet filtering approaches ineffective. Deep Packet Inspection enables network administrators to inspect protocol metadata and classify applications without decrypting payload data. This project introduces a lightweight DPI engine capable of parsing packets, extracting TLS SNI values, tracking flows using five-tuple identification, and enforcing filtering rules.

II. LITERATURE REVIEW

Traditional firewalls mainly operate at Layer 3 and Layer 4 of the OSI model. Modern DPI systems such as Snort and Suricata provide advanced traffic analysis features but require significant computational resources. Research on encrypted traffic classification highlights the use of TLS metadata such as Server Name Indication (SNI) for identifying applications without decryption. The proposed DPI engine combines educational simplicity with practical packet inspection functionality.

III. OBJECTIVES

The objectives of this project include traffic classification, TLS SNI extraction, application detection, flow tracking, rule-based filtering, and scalable multi-threaded packet processing.

IV. SYSTEM ARCHITECTURE

The DPI engine consists of modules including a PCAP reader, packet parser, flow tracker, SNI extractor, rule manager, and reporting engine. The multi-threaded version includes Reader Threads, Load Balancers, Fast Path processing threads, and Output Writer threads.

V. METHODOLOGY

The workflow begins with reading packets from PCAP files. The parser extracts Ethernet, IPv4, TCP, and UDP header information. Five-tuple identifiers are generated to track network flows. TLS Client Hello packets are inspected to extract SNI values, which are mapped to applications such as YouTube, Facebook, and Google. Filtering rules are then applied to determine whether packets should be forwarded or dropped.

VI. MULTI-THREADED PROCESSING

The DPI engine uses a producer-consumer architecture with thread-safe queues for parallel packet processing. Consistent hashing ensures that packets belonging to the same connection are processed by the same Fast Path thread. This architecture improves scalability and CPU utilization.

VII. EXPERIMENTAL RESULTS

Experimental testing demonstrated successful classification of encrypted traffic and effective application-level blocking. The engine correctly identified traffic associated with YouTube, Facebook, GitHub, and Google. The multi-threaded implementation distributed workloads efficiently across processing threads.

VIII. ADVANTAGES

The proposed system provides real-time traffic inspection, scalable packet processing, encrypted traffic analysis, flow-based filtering, and modular architecture suitable for cybersecurity research and education.

IX. LIMITATIONS

The system relies on TLS SNI visibility and may face challenges with encrypted SNI technologies such as ECH. Signature-based classification may also miss unknown applications.

X. FUTURE WORK

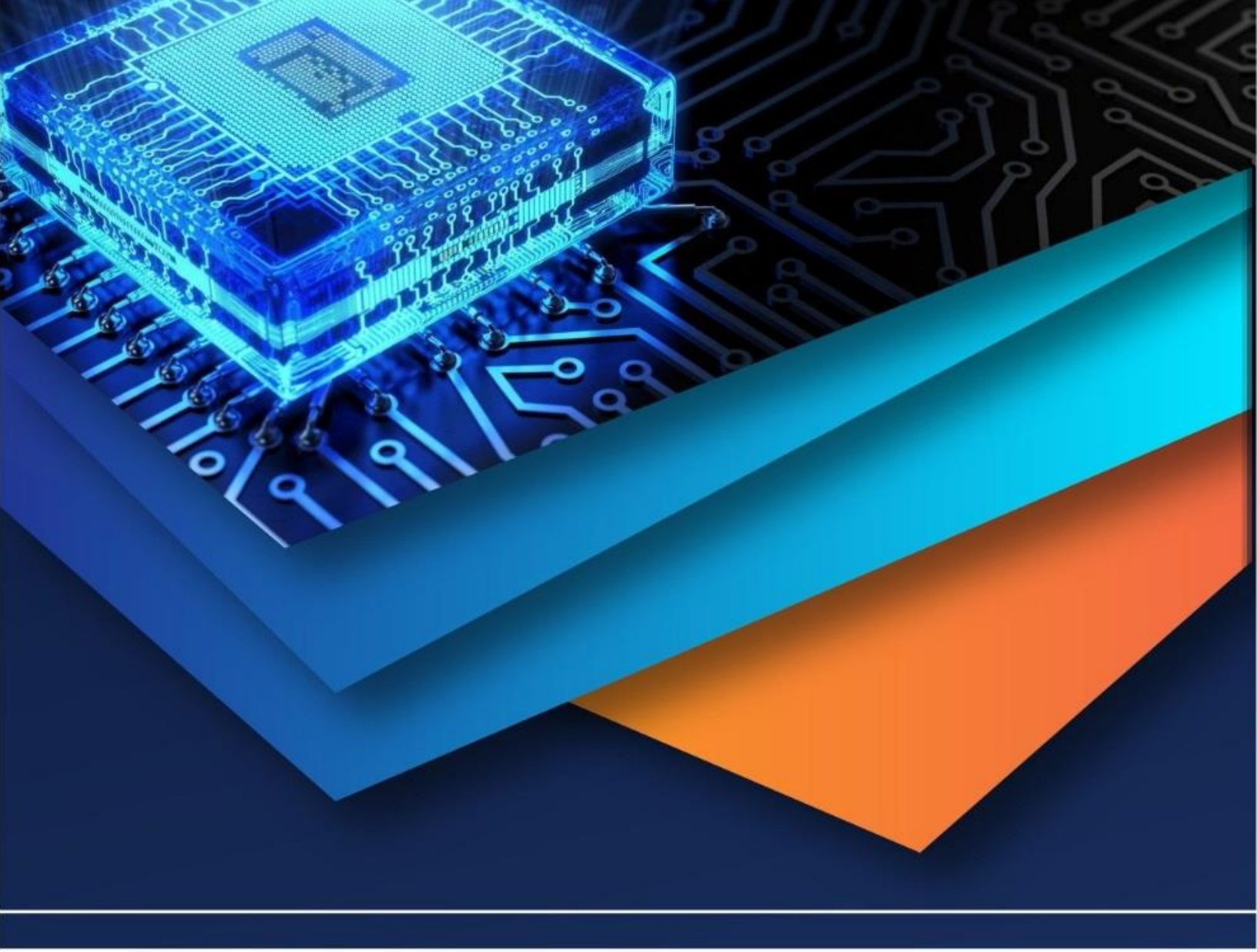
Future improvements include machine learning-based traffic classification, HTTP/3 and QUIC support, GPU acceleration, intrusion detection integration, and cloud-native deployment.

XI. CONCLUSION

This paper presented a Deep Packet Inspection engine for encrypted traffic analysis and application-level classification. The system demonstrates how TLS metadata can be used to identify and filter applications without decrypting payloads. The modular multi-threaded architecture makes the engine suitable for cybersecurity education, experimentation, and future enterprise-scale expansion.

REFERENCES

[1] RFC 6066 - TLS Extensions: Extension Definitions.
[2] RFC 8446 - TLS Protocol Version 1.3.
[3] Snort Intrusion Detection System.
[4] Suricata IDS/IPS Engine.
[5] Wireshark Network Protocol Analyzer.
[6] W. Stallings, Network Security Essentials.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY