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.2026.82553

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

Sanskar Maini
Centre of Artificial Intelligence, Madhav Institute of Technology & Science, India

Abstract: ***Modern computer networks generate massive volumes of internet traffic, making network monitoring and cybersecurity analysis increasingly important. Traditional packet filtering techniques based only on IP addresses and port numbers are insufficient for identifying modern encrypted internet applications. This paper presents the design and implementation of a multithreaded Deep Packet Inspection (DPI) framework for offline network traffic analysis. The proposed system processes packet capture (PCAP) files, extracts protocol-level information, analyzes TLS Server Name Indication (SNI) fields, and performs application-level traffic classification. A concurrent packet processing architecture using load balancer and worker threads is implemented to improve scalability and processing efficiency. The framework also supports application-based filtering and packet analysis for encrypted HTTPS traffic. Experimental results demonstrate successful packet classification and efficient multithreaded traffic processing suitable for educational and cybersecurity research applications.***

Keywords: Deep Packet Inspection, Network Security, Packet Analysis, TLS, SNI, Multithreading, Traffic Classification, PCAP, Cybersecurity, HTTPS Analysis

I. INTRODUCTION

Modern computer networks generate massive amounts of internet traffic, creating significant challenges for network monitoring, traffic management, and cybersecurity analysis. Traditional packet filtering techniques based only on IP addresses and port numbers are insufficient for identifying modern encrypted applications such as YouTube, Facebook, and HTTPS-based services. Deep Packet Inspection (DPI) enables advanced traffic analysis by examining packet contents and protocol-level metadata to classify applications and detect network activities more accurately.

This paper presents the design and implementation of a multithreaded Deep Packet Inspection engine for offline network traffic analysis. The proposed system processes packet capture (PCAP) files, parses network packets, extracts TLS Server Name Indication (SNI) information, and performs application-level traffic classification. A concurrent architecture using load balancer and fast-path worker threads is implemented to improve packet processing efficiency and scalability. The system also supports application-based filtering and packet blocking mechanisms, providing an effective framework for educational and cybersecurity research purposes.

II. RELATED WORK

Deep Packet Inspection (DPI) has become an important technique in modern network monitoring and cybersecurity systems for analyzing internet traffic beyond traditional packet filtering methods [1]. Traditional firewalls mainly rely on IP addresses and port numbers, which are insufficient for identifying encrypted internet applications [2]. Modern DPI systems utilize packet parsing, protocol inspection, and TLS SNI extraction techniques to identify encrypted services such as YouTube, Facebook, and HTTPS-based applications. Several packet analysis tools such as Wireshark and libpcap-based frameworks provide packet capture and protocol analysis functionalities for offline traffic inspection [3]. Recent research has also focused on improving packet processing efficiency using multithreaded and concurrent architectures for scalable traffic analysis. The proposed system follows a similar approach by implementing a multithreaded DPI framework for offline packet analysis, application classification, and traffic filtering using TLS SNI-based inspection techniques.

III. METHODOLOGY

The proposed system processes offline packet capture (PCAP) files and performs Deep Packet Inspection using a multithreaded packet processing framework. The methodology consists of packet parsing, TLS SNI extraction, application classification, and packet filtering stages.

Initially, packets are read from the PCAP file using the packet reader module. The packet parser extracts Ethernet, IPv4, TCP, and UDP header information along with payload data required for protocol analysis. TLS Client Hello packets are analyzed to extract the Server Name Indication (SNI) field, which is used to identify encrypted internet applications.

The extracted domain information is mapped to application categories such as YouTube, Facebook, HTTPS, and DNS services. A multithreaded architecture using worker threads and thread-safe queues improves packet processing efficiency and scalability. The system also supports application-based filtering and blocking policies and generates a filtered output PCAP file after processing.

IV. IMPLEMENTATION

The proposed Deep Packet Inspection framework is implemented using C++17 and multithreading libraries for efficient packet processing and traffic analysis. The system is developed in a modular manner, where separate modules handle packet reading, packet parsing, TLS inspection, traffic classification, and output generation.

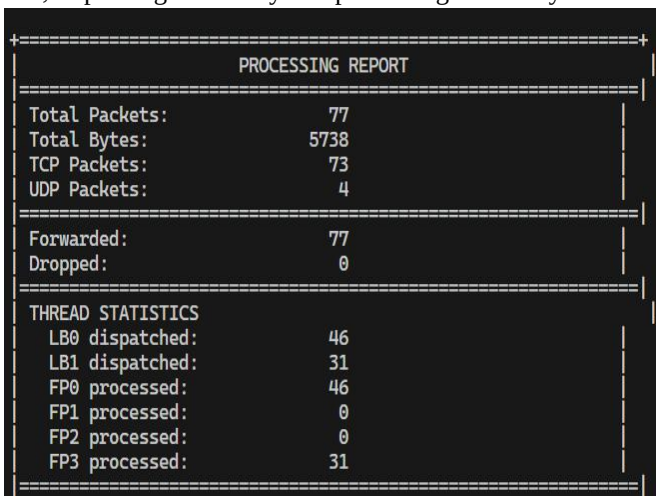
The packet reader module reads packets from offline PCAP files and forwards them to worker threads through thread-safe queues. The packet parser extracts protocol-level information including Ethernet, IPv4, TCP, and UDP headers. TLS Client Hello packets are analyzed by the SNI extractor module to obtain domain information from encrypted HTTPS traffic.

Application classification is performed using domain-based mapping techniques to identify internet applications such as YouTube, Facebook, HTTPS, and DNS traffic. A concurrent architecture using load balancer and fast-path worker threads improves scalability and packet processing throughput. The framework also supports application-based packet filtering and generates filtered output PCAP files after processing.

V. RESULTS AND ANALYSIS

The proposed Deep Packet Inspection engine was tested using offline packet capture (PCAP) files containing HTTPS, DNS, and application traffic. The framework successfully parsed packets, extracted TLS SNI information, classified internet applications, and generated processing statistics using a multithreaded architecture.

Experimental results showed that the system processed a total of 77 packets, including 73 TCP packets and 4 UDP packets. All packets were successfully forwarded without packet loss during execution. The concurrent packet processing framework distributed packets among multiple worker threads, improving scalability and processing efficiency.



```
+============================================================+
|                      PROCESSING REPORT                     |
|============================================================|
| Total Packets:                77                           |
| Total Bytes:                5738                           |
| TCP Packets:                  73                           |
| UDP Packets:                   4                           |
|============================================================|
| Forwarded:                    77                           |
| Dropped:                       0                           |
|============================================================|
| THREAD STATISTICS                                          |
|   LB0 dispatched:             46                           |
|   LB1 dispatched:             31                           |
|   FP0 processed:              46                           |
|   FP1 processed:               0                           |
|   FP2 processed:               0                           |
|   FP3 processed:              31                           |
|============================================================|
```

VI. CONCLUSION

This paper presented the design and implementation of a multithreaded Deep Packet Inspection framework for offline network traffic analysis and application classification. The proposed system successfully processes packet capture files, extracts protocol-level information, performs TLS SNI inspection, and classifies encrypted internet applications such as YouTube, Facebook, and HTTPS-based services.

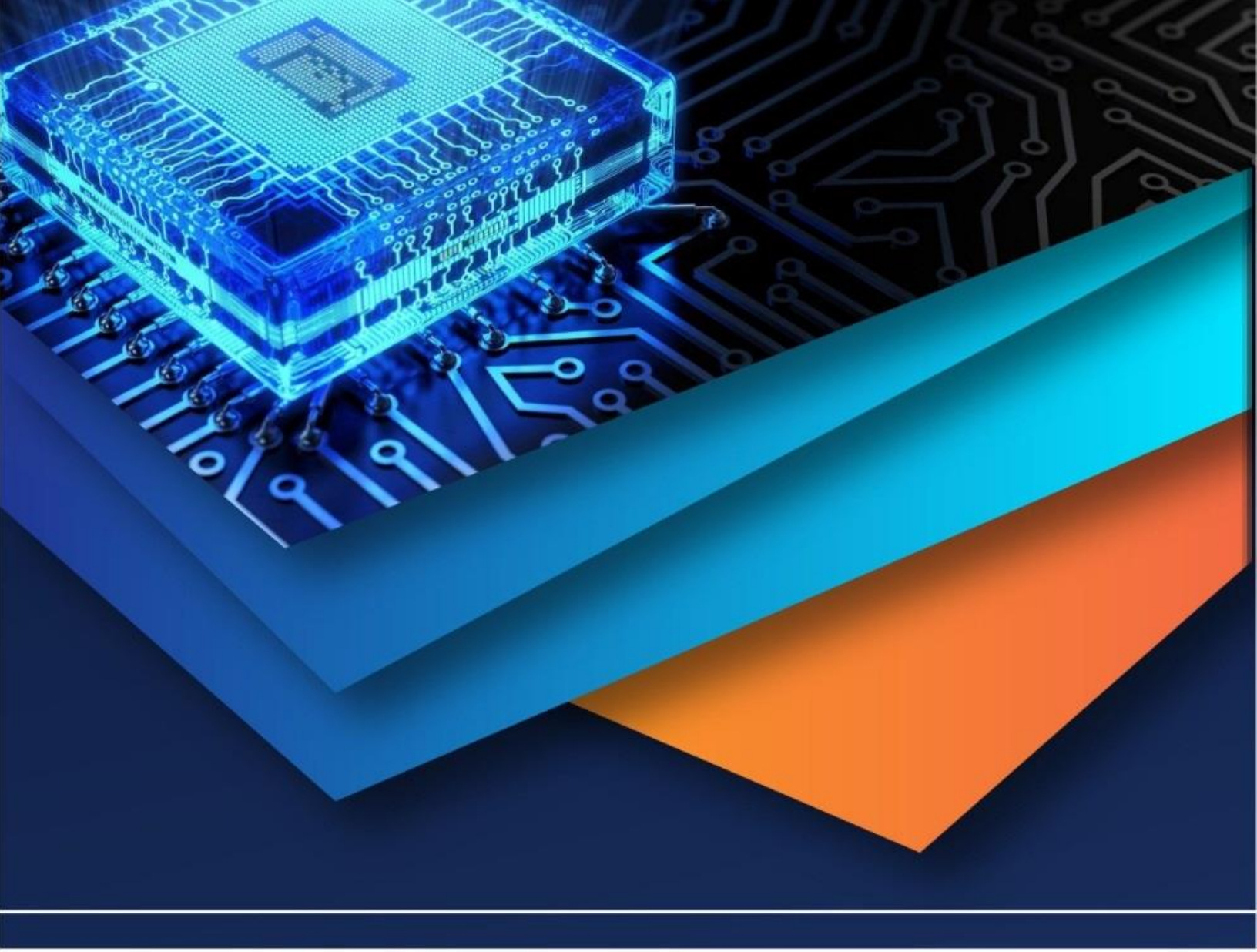
The implementation utilizes a concurrent packet processing architecture using load balancer and worker threads to improve scalability and processing efficiency. Experimental results demonstrated successful packet parsing, application detection, and traffic analysis using multithreaded execution. The proposed framework provides an effective solution for educational research, cybersecurity analysis, and network traffic monitoring applications.

VII. ACKNOWLEDGMENT

The author thanks the project supervisor and the Centre of Artificial Intelligence, Madhav Institute of Technology & Science, for their guidance and computational support throughout this work. This paper is based on a student academic project undertaken as part of the undergraduate program at the institute.

REFERENCES

[1] W. Stallings, Network Security Essentials: Applications and Standards, Pearson Education, 2017.
[2] RFC 6066, “Transport Layer Security (TLS) Extensions: Extension Definitions,” Internet Engineering Task Force.
[3] Wireshark Foundation, “Wireshark Network Protocol Analyzer,” Available: https://www.wireshark.org/



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY