International Journal of Engineering Trends and Technology
ISSN: 2231-5381/ https://doi.org/10.14445/22315381/IJETT-V74I11P114

Volume 74 Issue 11, 187-201, November 2026
© 2026 Seventh Sense Research Group®

Original Article

ALMDS-NDN: A Lightweight Adaptive Multi-Layer Defense Framework for Secure and Resilient Named Data Networking

Abdul Faiz. A[1], Sheelaselvakumari. N.A[2]

[1,2]*Department of Computer Science, Sri Krishna Arts and Science College, Coimbatore, Tamil Nadu, India.*

[1]*Corresponding Author: faiz.faizu015@gmail.com*

Received: 19 June 2026 Revised: 12 September 2026 Accepted: 18 September 2026 Published: 25 November 2026

Abstract *- The Named Data Networking (NDN) is an attractive data-centric networking solution that enhances content delivery through name-based routing and caching. Nevertheless, its stateful forwarding and caching processes make NDN prone to severe security risks such as Interest Flooding Attack (IFA), Cache Pollution Attack (CPA), and Cache Poisoning Attack (CPOA). This paper seeks to tackle these problems by introducing ALMDS-NDN, an Adaptive Lightweight Multi-Layer Defense System aimed at detecting, mitigating, and handling multiple attacks. ALMDS-NDN encompasses five complementary layers including a Monitoring layer based on ARIMA modeling for traffic analysis, a Detection and Analysis layer based on Isolation Forest and signature check, a Mitigation layer involving Adaptive Reputation-based Access Control (ARAC), tokens for rate limiting and priority cache, a Response layer implementing Dynamic Attack Isolation and Purging (DAIP), and finally Adaptive Data Selection Layer for efficient user data selection. Experiments conducted by deploying ndnSIM on a 75-node topology show that ALMDS-NDN can attain an interest satisfaction ratio of 99.6%, minimize average end-to-end delay 28 ms in a multi-attack situation, and boost the cache hit ratio to 91.4% while introducing only 3.8% extra CPU overhead relative to the traditional ELC algorithm. Moreover, the model proves resilient in scenarios where as high as 80% of the core routers are attacked, situations where other systems have proven ineffective.*

Keywords *- Named Data Networking (NDN), Interest Flooding Attack, Multi-Layer defense, Isolation forest, Adaptive Reputation-based Access Control.*

1. Introduction

Internet services and data-driven applications have proliferated to such an extent that the conventional host-oriented approach of networking becomes inadequate. In response to this trend, Named Data Networking (NDN), which constitutes part of the ICN paradigm, provides an interesting model of network architecture. Unlike other networking paradigms, ICN operates based on content names instead of IP host addresses [1, 2]. NDN uses Interest and Data packets, where the user sends Interest packets and receives the relevant Data packets from the network for efficient content dissemination, thanks to in-network caching, multicast transmission, and stateful forwarding.

Although NDN offers many benefits, it presents some security issues inherent in the structure itself. There are vulnerabilities associated with the PIT, CS, and FIB [3, 4]. Some of the most important among these attacks are Interest Flooding Attacks (IFA) that target the PIT and degrade its functioning through excessive Interest generation [9, 14]; Cache Pollution Attacks (CPA) that harm cache operation by adding unpopular or harmful content [6, 8]; and Cache Poisoning Attacks (CPOA) that lead to compromising data integrity through tampering with cache content [10, 11]. Several measures have been developed to overcome such security issues. Interest Flooding mitigation techniques include Satisfaction-Based Pushback (SBP) and Interest traceback, which restricts request flooding and malicious sources identification [12, 13]. Cache Pollution prevention tools include SLRU and CacheShield that filter out unpopular content to mitigate its harmful influence on cache operation [7, 11].

Digital signatures guarantee data authenticity, protecting from cache poisoning attacks [5]. However, current protection methods are rather narrow in scope as each of them addresses one problem at a time. Moreover, most proposed solutions introduce additional computational burden that limits their applicability in dynamically changing networks. To address such drawbacks, this research proposes a security mechanism called ALMDS-NDN, which is a multi-layer security mechanism that can adaptively deal with multiple attack


This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/)

vectors. The architecture of the proposed approach is multi-layered and includes the following techniques, namely real-time traffic monitoring, anomaly detection using machine learning, lightweight signature validation, access control and dynamic attacks isolation. Particularly, ARIMA-based traffic modeling is used to detect traffic patterns, whereas Isolation Forest is used for anomaly detection. The lightweight signature validation mechanism is applied to verify the suspicious data traffic to prevent unnecessary overheads.

Moreover, Adaptive Reputation-based Access Control (ARAC) mechanism, along with token regulation and caching are used to reduce malicious behaviours. In addition, Dynamic Attack Isolation and Purging mechanism isolates the compromised node and purges the poisoned content from the network. Adaptive User-Centric Data Selection mechanism is another layer included into this approach. In conclusion, the key advantages of the current work include:

- The development of a combined multi-layered protection solution for multiple NDN attacks.
- The use of machine learning and statistical approaches to timely detect any anomalies.
- The implementation of an adaptive reputation-based access control and caching approach; and
- The creation of a dynamic response solution for fast detection and isolation of attacks.

2. Related Work

With emerging threats like Interest Flooding Attacks, Cache Pollution Attacks, and Cache Poisoning attacks. Despite the data-centric nature of security in NDN, its architectural elements, including PIT, and in-network caching, have been found vulnerable to exploitation, which requires better defense strategies. According to Ullah et al. [14], various issues associated with existing approaches against cache poisoning attacks have been highlighted, including increased latency, increased computation time, and absence of any real-world testing. The authors point out that although cryptographic signature verifies the authenticity of information but performing complete verification on routers is not practical in high-throughput network scenarios. Therefore, researchers have looked forward toward light-weight verification processes.

According to Kumari et al. [15], an anomaly detection method has been introduced to detect interest flooding attacks. A statistical traffic analysis technique has been used to detect interest flooding attacks. Results demonstrate high efficiency of detection irrespective of the type of IFA attack variant. However, no mitigation technique has been considered for this model. Ogunbunmi et al. [16] discussed a comprehensive literature review on Interest Flooding attacks, indicating that exhaustion of the PIT table is the main factor that leads to failure to deliver service in NDN networks. Several methods and mitigation techniques are outlined, but the main conclusion is that current technologies use only static thresholds, making them less efficient in complex and dynamic networks. MaheshKumar et al. [17] suggested INI-24 dataset that simulates real scenarios for Interest Flooding attacks' assessment. Despite its great contribution, this technique cannot serve as a full-scale solution to the problem since it does not provide any measures of mitigation and protection against the attack.

Fan et al. [18] examined the potential use of NDN technologies in Internet of Vehicles (IoV). Their method allows one to ensure the safety of data communication by applying NDN-specific mechanisms of caching and naming; however, their method cannot be generalized to other problems related to network security in NDN environment. Despite all these recent advancements, some drawbacks still stand out. First, there is a lack of studies addressing multi-attack mitigation.

Also, most of the presented methods suffer from increased computational overhead and, thus, cannot be used in real-time applications. Moreover, none of the existing systems include any adaptivity features, which make their performance insufficient to cope with traffic changes. In addition, robustness regarding large-scale node failures is often not discussed. In order to address all of these shortcomings, an Adaptive Lightweight Multi-Layered Defense System (ALMDS) for NDN architecture based on statistical traffic modeling, machine learning techniques for anomaly detection, light-weight data packet verification, adaptive reputation-based access control and dynamic responses to attacks is presented. In contrast to current solutions, the suggested one is designed in a holistic way and incorporates all three types of attacks into consideration.

Current research has demonstrated notable progress achieved in NDN and ML-based multi-layer security schemes by taking into account aspects such as scalability, intelligence, and adaptation. Thus, Asif et al. [19] concentrated on the optimization of naming and packets lookup processes in NDN for increasing routing efficiency without any consideration of security issues.

In turn, Alsarhan et al. [20] introduced a concept of an AI-based multi-layer system aimed at real-time GPS spoofing detection, thus proving the high potential of multi-layer structures along with machine learning in such environments as UASs. Padmanaban et al. [21] introduced another multi-layer approach focused on adaptive multi-engine cyber defense with AI-based intelligent deception, whereas Alimohammadi et al. [22] presented a similar structure for providing protection against attacks on Open RAN infrastructure. Finally, Alimohammadi et al. [22] suggested applying advanced deep learning based on conditional GANs and vision transformers for federated predictions.

Nevertheless, all the above-mentioned models were specific for particular domains, failed to provide multi-attack protection, and caused additional computational expenses. This situation stimulates the further analysis of the suggested ALMDS-NDN model that incorporates multi-layer protection, machine learning, and adaptive control for the purpose of NDN environment protection.

Modern studies have seen a lot of research on adaptive and multi-layer security architectures in response to challenging cyber threats in various network environments. Zhao et al. [23] suggested a solution of adaptive multi-layer defense architecture for federated learning to be used in network security evaluation. This paper has shown the significance of applying a combination of multi-security layers along with adaptive decision-making algorithms.

Nevertheless, their architecture is built for federated learning environments and is not intended to tackle architectural challenges of NDN, including Interest flooding, Cache pollution, and Cache poisoning attacks. Moreover, Ullah et al. [24] have developed an intelligent multi-layered architecture for autonomous cyber threat detection and adaptive protection for cloud-native smart electrical power grid cyber-physical system infrastructure using machine learning and big data technologies. Even though this paper presents promising results of using an intelligent multi-layer security architecture, it is designed for a specific domain - cyber-physical power grid systems.

In particular, in relation to the current research domain, Mirajkar et al. [25] designed adaptive attack-resistant algorithm and orchestration of security solutions for Named Data Networking. The above study brings into focus the increasing significance of adaptive security techniques in NDN scenarios. However, unlike the current study, the ALMDS-NDN framework incorporates statistical traffic analysis, machine learning-based anomaly detection, signature-based authentication, adaptive reputation-based access control, priority based caching and dynamic attack isolation and purging under one umbrella to combat multiple types of attacks.

In addition, Kosuri et al. [26] put forth a lightweight multi-layer security framework for IoT devices, wherein they have highlighted the need for adaptive and scalable security solutions in the face of new threats in connected systems. Similar to the present study, the concepts of lightweight and multi-layer security solutions have been utilized, however the framework has been proposed for IoT devices and not for NDN.

Moreover, Mustafa et al. [27] propose a secure and energy-efficient cross-layer architecture design for the Internet of Things, stressing on the importance of security along with network efficiency in a number of architectural layers. This shows the importance of multi-layer security mechanisms; nevertheless, it is focused on IoT communications and energy efficiency rather than detection and mitigation of Interest Flooding, Cache Pollution, and Cache Poisoning attacks in NDN simultaneously. Thus, while these are indeed important advances that have been made in the field of security mechanisms for network layers, there still exists the necessity for a lightweight and adaptive approach aimed at multi-attack detection and mitigation in Named Data Networking. This is the void that is covered by the presented in this paper ALMDS-NDN approach.

2.1. Research Gap

Despite various studies that have provided solutions to Interest Flooding, Cache Pollution, and Cache Poisoning attacks individually or by considering certain security issues, such as attack detection or mitigation, the methods have been found to be limited by either using fixed thresholds or adding computation overheads, making them less adaptable in NDN networks.

There have also been recent developments of adaptive multi-layer security approaches for federated learning, Internet of Things, and cyber-physical systems; however, these are yet to provide security for the vulnerability arising from PIT exhaustion, cache pollution, and content poisoning in NDN.

There is therefore a need for an adaptive, lightweight, and integrated approach that can detect and mitigate the different NDN attacks in real-time. This paper presents an Adaptive Lightweight Multi-Layer Defense System for NDN (ALMDS-NDN), which uses a five-layer approach combining ARIMA-based monitoring, Isolation Forest-based anomaly detection, lightweight signature verification, Adaptive Reputation-based Access Control, Priority-Based Caching, and Dynamic Attack Isolation and Purging.

3. Proposed Methodology

The design of the suggested Adaptive Lightweight Multi-Layer Defense System for Named Data Networking (ALMDS-NDN) can be regarded as an effort to create a system capable of dealing with a number of security issues associated with NDN by detecting, analyzing, mitigating, and responding to threats such as Interest Flooding Attacks, Cache Pollution Attacks, and Cache Poisoning Attacks. The system uses a multi-layer approach that employs statistical modeling, machine learning, adaptive control, and dynamic response methods.

3.1. System Architecture

The proposed ALMDS-NDN architecture is a multi-layer architecture that guarantees secure and efficient data transmission in Named Data Networking through the use of Monitoring, Detection and Analysis, Mitigation, Response,

and Adaptive Data Selection Layers. The architecture starts by gathering the network traffic consisting of both normal and attack cases and then analyzing it in the Monitoring layer via ARIMA-based profiling and content access logs to define the norm. Next, the Detection and Analysis layer detects anomalies through Isolation Forest, verifies data signatures for integrity and uses behavior scores to determine malicious nodes.

The Mitigation layer manages suspicious activities using token management, reputation-based access control and priority-based caching to prevent resource abuse. The Response layer carries out dynamic attack isolation and cache eviction operations with the help of alerting capabilities. Finally, the Adaptive Data Selection layer facilitates secure and fast data fetching from caches by users using their priorities.

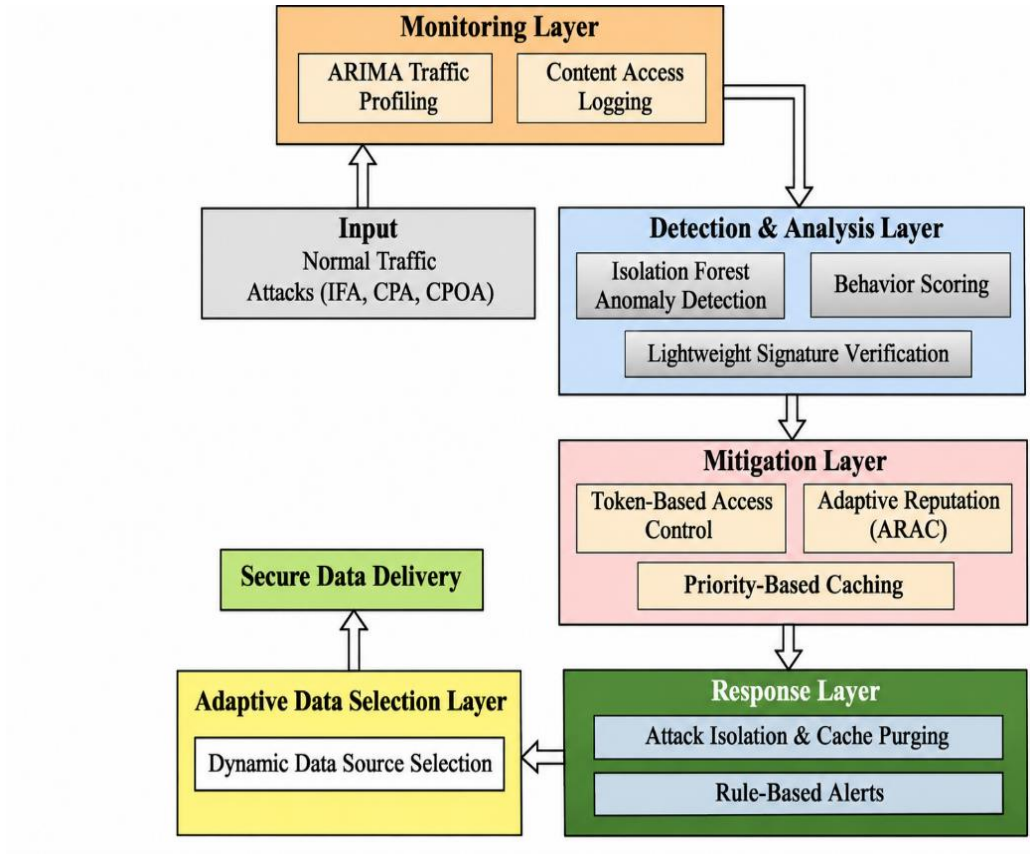


Fig. 1 Architecture of the proposed ALMDS-NDN framework

The architecture of the proposed framework of ALMDS-NDN is illustrated in Figure 1, wherein a multi-layered approach is applied to monitor, detect, and counteract various attacks on NDN in order to maintain effective and secure communication of data packets.

3.2. Monitoring Layer

Monitoring Layer of ALMDS-NDN is tasked with constant monitoring of network traffic behavior, forming a statistical model that characterizes normal operation of network infrastructure. This module uses such indicators as Interest packet arrival rate, number of requests issued, degree of content popularity, and latency for traffic anomaly detection.

This information is necessary to differentiate between usual network traffic and unusual one resulting from attacks such as Interest Flooding and Cache Pollution. As part of temporal analysis of network traffic dynamics, the use of Autoregressive Integrated Moving Average (ARIMA) models is proposed. The ARIMA model can be characterized as one of the most used algorithms in time-series forecasting.

ARIMA provides a good approximation for modeling dependencies on different scales, making it possible to predict network traffic behavior accurately.

The general ARIMA model is denoted as ARIMA (p, d, q), where p represents the autoregressive order, d the degree of differencing, and q the moving average order. The mathematical formulation is given in Equation (1):

$$X_t = c + \sum_{i=1}^{p} \phi_i X_{t-i} + \sum_{j=1}^{q} \theta_j \epsilon_{t-j} + \epsilon_t \qquad (1)$$

Where $X_t$represents the observed traffic at time t, $c$is a constant, $\phi_i$are autoregressive coefficients, $\theta_j$are moving average coefficients, and $\epsilon_t$is white noise. To ensure stationarity, differencing is applied as using Equation (2):

$$Y_t = X_t - X_{t-1} \tag{2}$$

Where $Y_t$is the differenced time series used for modeling. The predicted traffic value $\hat{X}_t$is computed using the trained ARIMA model, and anomaly detection is performed by calculating the deviation between observed and predicted values:

$$D_t = | X_t - \hat{X}_t | \tag{3}$$

If the deviation exceeds a predefined threshold δ, the traffic is considered anomalous:

$$D_t > \delta \tag{4}$$

In addition to traffic modeling, the Monitoring Layer evaluates content popularity, which is critical for identifying cache pollution attacks. The popularity score of a content item is calculated as:

$$P_i = \frac{R_i}{\sum_{k=1}^{N} R_k} \tag{5}$$

Where R_i is the number of requests for content *i*, and *N* is the total number of content items. Furthermore, request rate monitoring is performed to detect abnormal surges in Interest packets:

$$\lambda_t = \frac{N_t}{\Delta t} \tag{6}$$

Where λ_t is the request rate, $N_t$is the number of requests in time interval Δt. -Through ARIMA traffic prediction, analysis of deviations, estimation of content popularity, and observation of the request rate, the Monitoring Layer can offer a strong basis for abnormality detection. This helps in ensuring that any potential attacks are detected early on without being too computationally intensive.

3.3. Detection and Analysis Layer

The Detection and Analysis Layer have the responsibility of detecting unusual network activities and the presence of attacks like Interest Flooding and Cache Pollution. The layer uses Isolation Forest (iForest), a machine learning technique used for anomaly detection in high-dimensional data sets.

As opposed to distance-based or density-based approaches that use tree generation, Isolation Forest uses isolation in order to isolate anomalies within the dataset. In ALMDS-NDN, each network node is represented by a feature vector:

$$\mathrm{x} = [r, \lambda, l, p] \tag{7}$$

Where r denotes request rate, λ represents traffic intensity, l is average latency, and $p$is content popularity score. These features are extracted from the Monitoring Layer and used as input for anomaly detection. Isolation Forest constructs multiple random binary trees (isolation trees) by recursively selecting a feature and splitting value. The number of splits required to isolate a data point corresponds to its path length, denoted as $h(x)$. Anomalies tend to have shorter path lengths since they are easier to isolate. The anomaly score for a data point $x$is computed as:

$$s(x, n) = 2^{-\frac{E(h(x))}{c(n)}} \tag{8}$$

Where $E(h(x))$ is the average path length of point $x$across all trees, $n$is the number of samples, and $c(n)$ is the normalization factor defined as:

$$c(n) = 2H(n-1) - \frac{2(n-1)}{n} \tag{9}$$

Here, $H(n)$ is the harmonic number, approximated as:

$$H(n) \approx \ln(n) + \gamma \tag{10}$$

Where $\gamma \approx 0.5772$ is the Euler-Mascheroni constant. The anomaly score $s(x, n)$lies in the range (0,1). A value close to 1 indicates a high probability of anomaly, while values near 0 indicate normal behavior. A threshold $\tau$is defined to classify anomalies:

$$s(x, n) > \tau \tag{11}$$

If the anomaly score exceeds the threshold, the node is classified as suspicious and forwarded to the mitigation layer. As can be seen from Figure 2, traffic of users and attackers is initially processed by the monitor layer. Traffic-related factors, including traffic volume and integrity, are extracted. The features are used as input for the detection layer. Using Isolation Forest and behavior scoring, anomaly detection is performed, after which the attacks are classified into respective types.

In addition to anomaly detection, a Behavior Scoring (BS) mechanism assigns a risk score to each node by combining multiple parameters:

$$S_i = w_1 r_i + w_2 \lambda_i + w_3 l_i + w_4 (1 - p_i) \tag{12}$$

Where S_i is the behavior score of node$i$, $w_1, w_2, w_3, w_4$are weighting factors, and the term $(1-p_i)$penalizes low-popularity (potentially malicious) content requests. To detect Cache Poisoning Attacks, the layer incorporates Lightweight

Signature Verification (LSV), where only suspicious data packets identified by anomaly detection are verified:

$$V(d) = \begin{cases} 1, & \text{if signature is valid} \\ 0, & \text{otherwise} \end{cases} \quad (13)$$

If the packets cannot pass through the validation process, then they are classified as being malicious, and they are sent to the response layer. Through the combination of Isolation Forest-based anomaly detection, behavioral score, and signature validation, this layer is able to ensure that the detection has high accuracy but low computational costs.

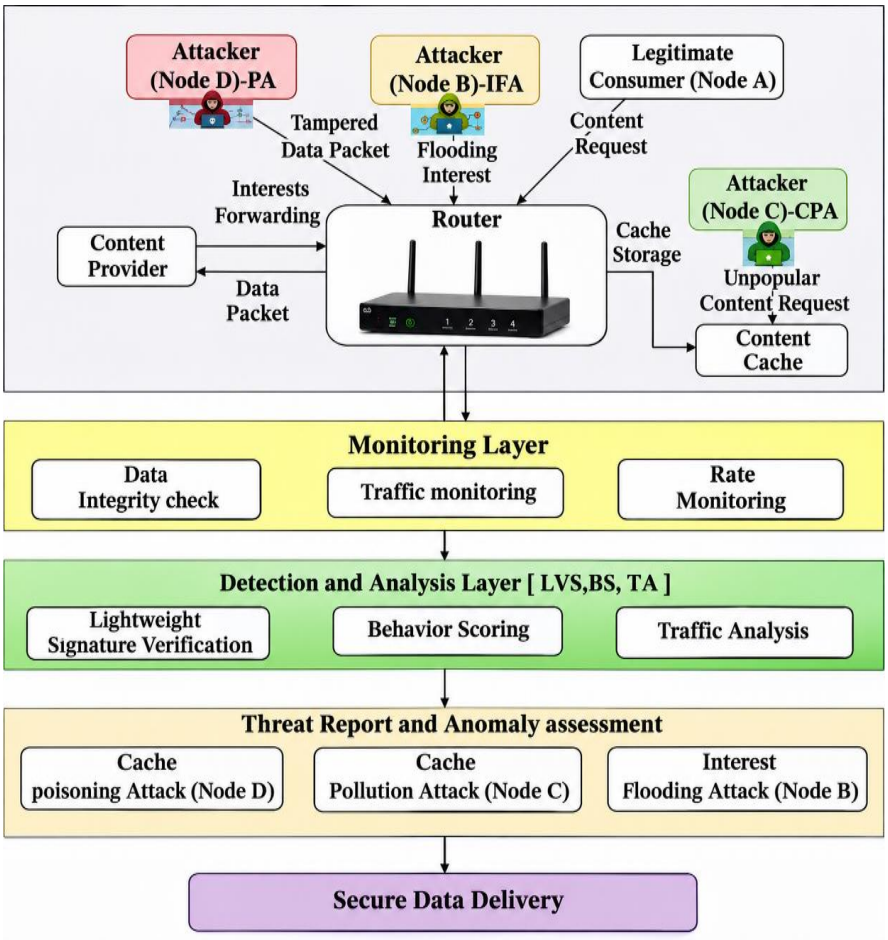


Fig. 2 Attack detection and analysis process in ALMDS-NDN

3.4. Mitigation Layer

Mitigation layer of ALMDS-NDN works for managing and minimizing the damage caused by any malicious node, but at the same time ensures that the system functions properly. The Mitigation layer uses Token-based Access control, Adaptive Reputation-based Access Control (ARAC), and Priority-Based caching. These features make sure that the resources of the system are utilized efficiently and also protect the system from attacks such as Interest Flooding and Cache Pollution.

3.4.1. Adaptive Reputation-based Access Control (ARAC)

Adaptive Reputation-based Access Control (ARAC) assesses the trustworthiness of each node depending upon its behavior and generates a reputation score for it. The reputation score of node*i* at time *t* is defined as:

$$R_i(t) = \alpha R_i(t-1) + (1-\alpha)(1-S_i) \quad (14)$$

where $R_i(t)$ is the updated reputation score, $R_i(t-1)$ is the previous reputation, $S_i$is the behavior score obtained from the Detection Layer, and $\alpha \in [0,1]$is a smoothing factor controlling the influence of past behavior.

Nodes with higher anomaly scores (higher S_i) receive lower reputation values, indicating potential malicious behavior.

3.4.2. Token-based Access Control

To regulate network traffic, each node is assigned a token value based on its reputation score. The number of allowed Interest packets is proportional to its trust level:

$$T_i = T_{max} \cdot R_i(t) \quad (15)$$

Where T_i is the number of tokens assigned to node *i*, and $T_{max}$is the maximum allowable token limit. The forwarding condition is defined as:

$$F_i = \begin{cases} 1, & \text{if } T_i > 0 \\ 0, & \text{otherwise} \end{cases} \quad (16)$$

Where $F_i = 1$allows forwarding of Interest packets, and $F_i = 0$blocks or delays requests from suspicious nodes.

3.4.3. Reputation Update with Feedback

To ensure adaptability, the reputation score is further refined using feedback from successful and failed transactions:

$$R_i(t) = R_i(t) + \beta(SR_i - FR_i) \quad (17)$$

where $SR_i$is the success rate (valid responses), $FR_i$is the failure rate (invalid or malicious activity), and $\beta$is a learning rate parameter.

3.4.4. Priority-based Probabilistic Caching

To mitigate Cache Pollution, the system implements a probabilistic caching strategy based on content popularity:

$$P_{cache}(i) = \frac{P_i}{\sum_{k=1}^{N} P_k} \quad (18)$$

Where $P_i$is the popularity score of content *i*. Frequently requested content has a higher probability of being cached, while low-popularity (potentially malicious) content is less likely to occupy cache space.

3.4.5. Mitigation Decision Rule

The Response layer in ALMDS-NDN plays the role of taking necessary actions in the event that any malicious behavior is detected using the Detection and Mitigation layers. The layer uses Dynamic Attack Isolation and Purging (DAIP) that entails isolating the nodes, clearing out caches, and raising alerts.

Through such measures, malicious traffic spread within the network is limited while at the same time ensuring that cached content remains intact. Malicious nodes are isolated according to their reputation and anomaly scores. A node is considered malicious and subjected to strict control if:

$$R_i(t) < \theta \quad (19)$$

Where $\theta$is the reputation threshold. Nodes below this threshold experience reduced tokens, limited forwarding capability, or are forwarded to the Response Layer for isolation.

3.5. Response Layer

The isolation decision is based on both reputation and anomaly thresholds:

$$I_i = \begin{cases} 1, & \text{if } R_i(t) < \theta \text{ or } s(x_i) > \tau \\ 0, & \text{otherwise} \end{cases} \quad (20)$$

Where $I_i$indicates isolation status of node *i*, $R_i(t)$ is the reputation score, $\theta$is the reputation threshold, $s(x_i)$ is the anomaly score, and $\tau$is the anomaly threshold. If $I_i = 1$, the node is blocked from forwarding or receiving packets.

To mitigate Cache Poisoning Attacks, suspicious or invalid content is removed from the cache using a purging function based on verification results and anomaly indicators:

$$C_i = \begin{cases} 0, & \text{if } V(d_i) = 0 \text{ or } s(x_i) > \tau \\ 1, & \text{otherwise} \end{cases} \quad (21)$$

Where $C_i$represents the cache validity status of content *i*, and $V(d_i)$ is the result of lightweight signature verification. If $C_i = 0$, the content is removed from the cache.

The system computes an attack severity score to prioritize response actions:

$$A_i = \gamma_1 s(x_i) + \gamma_2(1 - R_i(t)) \quad (22)$$

Where $A_i$is the severity score, $\gamma_1$and $\gamma_2$are weighting factors. Higher values indicate more severe attacks requiring immediate isolation and aggressive mitigation.

As depicted in Figure 3, the suggested framework combines the mitigation strategy based on ARAC with dynamic response techniques such as node isolation, traffic filtering, and cache eviction, which is succeeded by data selection.

A Rule-Based Alert System logs detected events and notifies administrators when abnormal behavior is confirmed:

$$Alert_i = \begin{cases} 1, & \text{if } A_i > \eta \\ 0, & \text{otherwise} \end{cases} \quad (23)$$

Where η is the alert threshold. If triggered, the system records the event details and generates a notification for monitoring and analysis. To avoid permanent exclusion of nodes due to temporary anomalies, a recovery mechanism is included:

$$R_i(t+1) = R_i(t) + \delta \quad (24)$$

Where δ is a recovery factor applied when a node resumes normal behavior. This allows previously isolated nodes to be gradually reintegrated into the network.

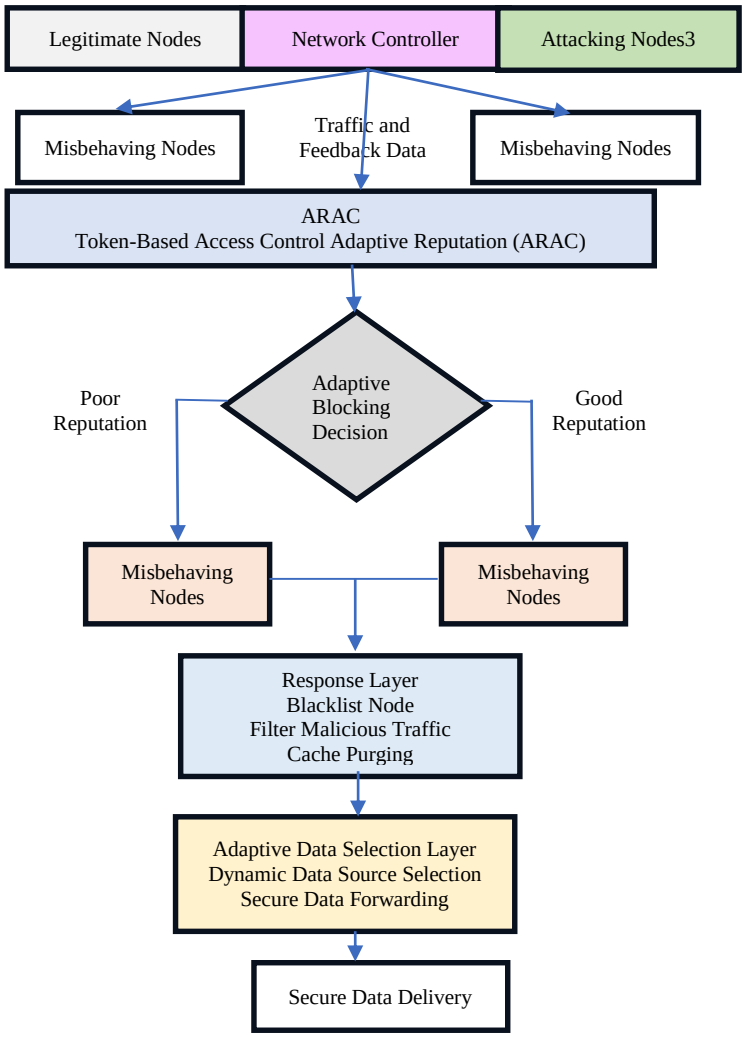


Fig. 3 Mitigation and response framework of ALMDS-NDN

3.6. Adaptive Data Selection Layer

ADSL of ALMDS-NDN is capable of making data access contextual and personalized through the selection of an appropriate data source, taking into account various parameters, including security and latency. The ADSL provides an ideal combination of QoS and security that can be selected depending upon whether speed or integrity needs to be preferred by the users. Each available data source $j$is evaluated using a composite score that integrates security and performance metrics:

$$D_j = w_s S_j + w_l \left(1 - \frac{L_j}{L_{max}}\right) \quad (25)$$

Where $D_j$is the selection score of source$j$, $S_j$is the security score (derived from reputation), $L_j$is the latency, $L_{max}$is the maximum observed latency for normalization, and $w_s, w_l$are weighting factors such that $w_s + w_l = 1$. The security score of a data source is derived from its reputation value:

$$S_j = R_j(t) \quad (26)$$

Where $R_j(t)$ is the reputation score obtained from the Mitigation Layer. Sources with higher reputation are considered more trustworthy. To ensure fair comparison across multiple sources, latency is normalized as:

$$L_j^{norm} = \frac{L_j}{L_{max}} \quad (27)$$

This ensures that latency values are scaled between 0 and 1. The system selects the optimal data source based on the maximum composite score:

$$j^* = \arg\max_j D_j \quad (28)$$

Where j^*represents the selected data provider. User preferences and network conditions influence the weighting factors dynamically:

$$w_s = \frac{U_s}{U_s+U_l}, w_l = \frac{U_l}{U_s+U_l} \quad (29)$$

Where U_s and U_l represent user-defined priorities for security and latency, respectively. This allows the system to adapt in real time based on application requirements. To prevent overloading a single data source, a load-balancing factor is incorporated:

$$D_j' = D_j - \lambda \cdot Load_j \quad (30)$$

Where $Load_j$ represents the current load on source j, and λ is a penalty factor.

This ensures balanced utilization of network resources.

Algorithm 1: Adaptive Lightweight Multi-Layer Defense System (ALMDS-NDN)

Input: Network traffic stream T(Interest packets, Data packets)
Output: Secure data delivery with detected and mitigated attacks

Step 1: Initialization

1. Initialize parameters: thresholds $\delta, \tau, \theta, \eta$
2. Initialize ARIMA model for traffic profiling
3. Initialize Isolation Forest model
4. Set initial reputation $R_i(0) = 1$for all nodes
5. Set token limit T_{max}

Step 2: Monitoring Layer

6. For each time interval t:
7. Collect traffic features: request rate r_i, latency l_i, content requests R_i
8. Predict expected traffic $\hat{X}_t$using ARIMA
9. Compute deviation: $D_t = \mid X_t - \hat{X}_t \mid$
10. Compute request rate: $\lambda_i = \frac{N_i}{\Delta t}$
11. Compute content popularity: $P_i = \frac{R_i}{\sum R_k}$
12. If $D_t > \delta$, mark traffic as suspicious

Step 3: Detection and Analysis Layer

13. Construct feature vector: $x_i = [r_i, \lambda_i, l_i, P_i]$
14. Compute anomaly score using Isolation Forest: $s(x_i)$
15. Compute behavior score:
    $S_i = w_1 r_i + w_2 \lambda_i + w_3 l_i + w_4(1 - P_i)$
16. If $s(x_i) > \tau$, mark node as anomalous
17. Perform Lightweight Signature Verification for suspicious data
18. If verification fails, mark data as malicious

Step 4: Mitigation Layer (ARAC + Token Control)

19. Update reputation score:
    $R_i(t) = \alpha R_i(t-1) + (1-\alpha)(1 - S_i)$
20. Assign tokens: $T_i = T_{max} \cdot R_i(t)$
21. If $T_i > 0$, allow packet forwarding; else restrict node
22. Update reputation with feedback:
    $R_i(t) = R_i(t) + \beta(SR_i - FR_i)$
23. Compute caching probability:
    $P_{cache}(i) = \frac{P_i}{\sum P_k}$

Step 5: Response Layer (DAIP)

24. If $R_i(t) < \theta$ or $s(x_i) > \tau$:
    Isolate node i
25. If data fails verification:
    Remove content from cache
26. Compute attack severity:
    $A_i = \gamma_1 s(x_i) + \gamma_2(1 - R_i(t))$
27. If $A_i > \eta$, generate alert
28. Apply recovery: $R_i(t+1) = R_i(t) + \delta$ (if behavior normalizes)

Step 6: Adaptive Data Selection Layer

29. For each data source j:
30. Compute selection score:
    $D_j = w_s S_j + w_l(1 - \frac{L_j}{L_{max}})$
31. Adjust with load factor:
    $D_j' = D_j - \lambda \cdot Load_j$
32. Select optimal source: $j^* = \arg\max D_j'$

Step 7: Output

33. Deliver data from selected source j^*
34. Ensure secure, efficient, and attack-resilient communication

End Algorithm

4. Experimental Setup

Experimental assessment of the proposed ALMDS-NDN architecture was performed through ndnSIM – a widely-used network simulation environment for modeling NDN systems.

For this experiment, a moderate-sized topology was assumed, involving 75 network nodes with the presence of consumer nodes, producer nodes, and intermediate routers supporting caching capability.

The simulation involved traffic generated under normal circumstances as well as traffic initiated via attacks on the network – Interest Flooding Attack (IFA), Cache Pollution Attack (CPA), and Cache Poisoning Attack (CPOA). In the experiment, the network traffic data, such as request rate, packet latency, access rate of cache, packet delivery information, and popularity of content in the network, were collected and used as inputs for the proposed modules.

For the implementation, the Python programming language was used for ARIMA traffic profile generation and Isolation Forest-based anomaly detection, ndnSIM, involving C++ programming language, was used for simulating the network environment, and SQLite was used for recording network events.

To guarantee feasibility in lightweight scenarios, the proposed system was experimented with on a computer that had an Intel Dual Core CPU, 2GB RAM, and 100GB disk space and was using Ubuntu version 20.04 and above. Development was done in the aid of development tools like Visual Studio Code and PyCharm.

The performance of the ALMDS-NDN approach was measured using the following standard parameters: True Positive Rate (TPR), False Positive Rate (FPR), Precision, Recall, F1-score, latency, throughput, packet loss ratio, cache hit ratio, CPU utilization, and Memory Utilization.

In order to make fair comparisons among systems, the performance of the proposed system was compared against existing systems like Satisfaction-Based Pushback (SBP), Segment Least Recently Used (SLRU), and Effective Lightweight Countermeasure (ELC).

5. Result and Discussion

Performance of the suggested ALMDS-NDN scheme is assessed under normal circumstances and under multiple attack conditions using ndnSIM. Performance assessment is done by comparing ALMDS-NDN scheme to other techniques like SBP, SLRU, and ELC, among others.

5.1. Quantitative Performance Evaluation

From Table 1, ALMDS-NDN surpasses other methodologies in performance. First, ISR is 99.6% in the case of the current methodology, which proves that the performance of content distribution is very reliable in spite of attacks on the network. Secondly, end-to-end delay is only 28ms, implying efficient management of congestion. Moreover, CHR has been found to be 91.4%, suggesting that cache utilization efficiency has been improved. However, there is an increase in CPU overhead to 3.8%.

Table 1. Performance comparison of ALMDS-NDN with baseline methods

Method	ISR (%)	Delay (ms)	CHR (%)	CPU Overhead (%)
SBP	82.4	65	70.2	2.1
SLRU	85.7	58	78.5	2.8
ELC	93.2	42	85.6	3.2
ALMDS-NDN	99.6	28	91.4	3.8

The effectiveness of ALMDS-NDN can be explained by its adaptive multilayer architecture that differs from both SBP and SLRU that are single-layer approaches. While SBP is intended mostly to counter Interest flooding, and SLRU deals with cache pollution problem, ALMDS-NDN simultaneously solves issues related to anomalies in the traffic, existence of malicious nodes, cache pollution, and poisoned content.

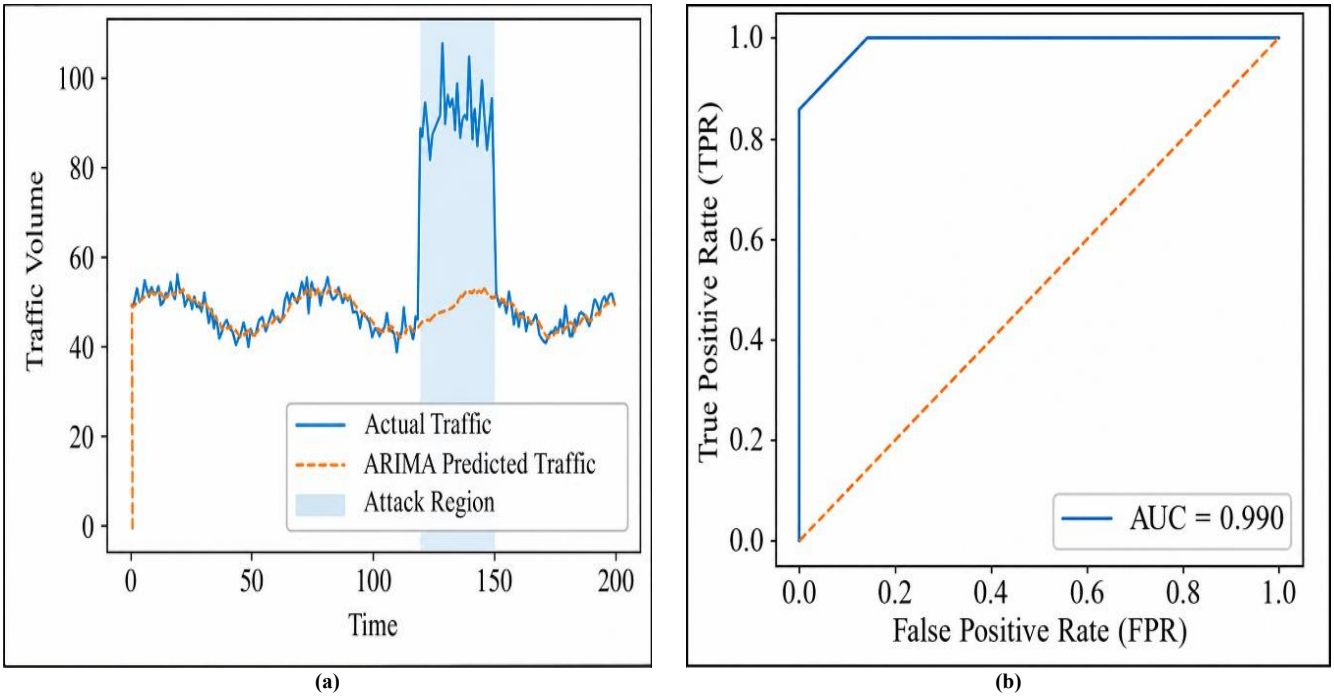
Traffic anomalies are detected using the ARIMA-based monitoring layer, Isolation Forest detects anomalies efficiently without the need for specific models for each particular attack type. Moreover, ARAC and token control regulate traffic of suspicious nodes according to their behavior and reduce the traffic and improve the ISR. Priority-based caching mechanism prevents malicious low-popularity content from occupying caches and increasing the CHR. And DAIP isolates malicious nodes quickly and eliminates poisoned cached content.

The synergy of these mechanisms leads to the improved results of the ALMDS-NDN approach that shows 99.6% ISR, 28 ms delay and 91.4% CHR. Despite the fact that the suggested framework has somewhat increased CPU overhead compared with the baseline approaches, it is not significant because expensive signature validation is done only for suspicious packets.

5.2. Detection and Monitoring Performance

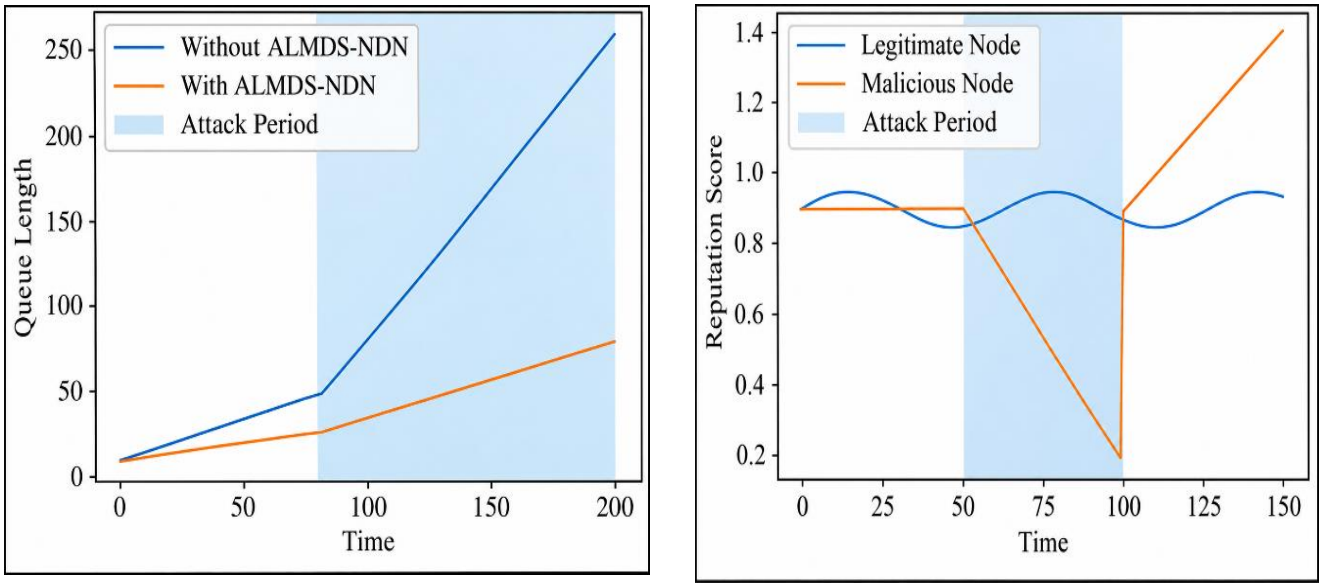
Detection capability of ALMDS-NDN is demonstrated in Figure 4(a), where the AUC value of the ROC curve is 0.99, implying high detection accuracy of the model. This proves the efficacy of the Isolation Forest model.

Monitoring capability based on the ARIMA model is shown in Figure 4(b). It can be noted that there is a visible discrepancy between the predicted and real values of traffic volume during attack.



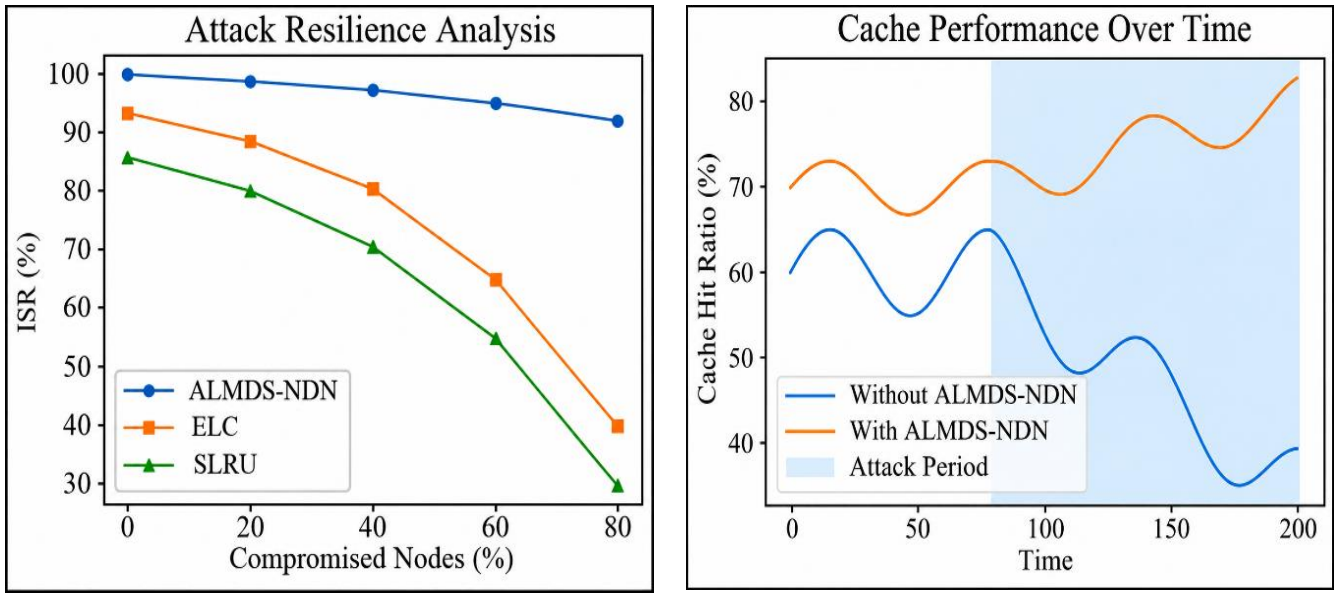
(a) **(b)**

Fig. 4 (a) ROC curve showing attack detection accuracy, and (b) ARIMA-based traffic profiling showing deviation during attack conditions.



(a) (b)

Fig. 5 (a) Queue length comparison under Interest Flooding Attack, and (b) Reputation score evolution using ARAC.



(a) (b)

Fig. 6 (a) Cache performance over time, and (b) Cache hit ratio comparison.

5.3. Congestion Control and Mitigation Analysis

From Figure 5(a), the length of the queue grows fast without any mitigation measures. This is due to the congestion that occurs due to Interest Flooding, but ALMDS-NDN manages to stabilize the queue with the help of tokens. Figure 5(b) shows how the ARAC approach works. Reputation scores for malicious nodes tend to decrease significantly during the attack stage, while the scores for the legitimate nodes remain constant.

5.4. Cache Performance Analysis

According to the results in Figure 6 (a), it can be seen that there is considerable performance reduction when cache is attacked. On the contrary, cache performance with respect to ALMDS-NDN has always been stable, thanks to its priority-based caching mechanism. As it is clear from Figure 6 (b), ALMDS-NDN outperforms others regarding CHR.

5.5. Overall Performance Comparison

From Figure 7, one can easily see that there is overall improvement by using ALMDS-NDN against other methods, such as SBP, SLRU, and ELC in terms of ISR, delay, and CHR.

From Figure 8, the reader can understand that ALMDS-NDN offers better performance with lower CPU usage.

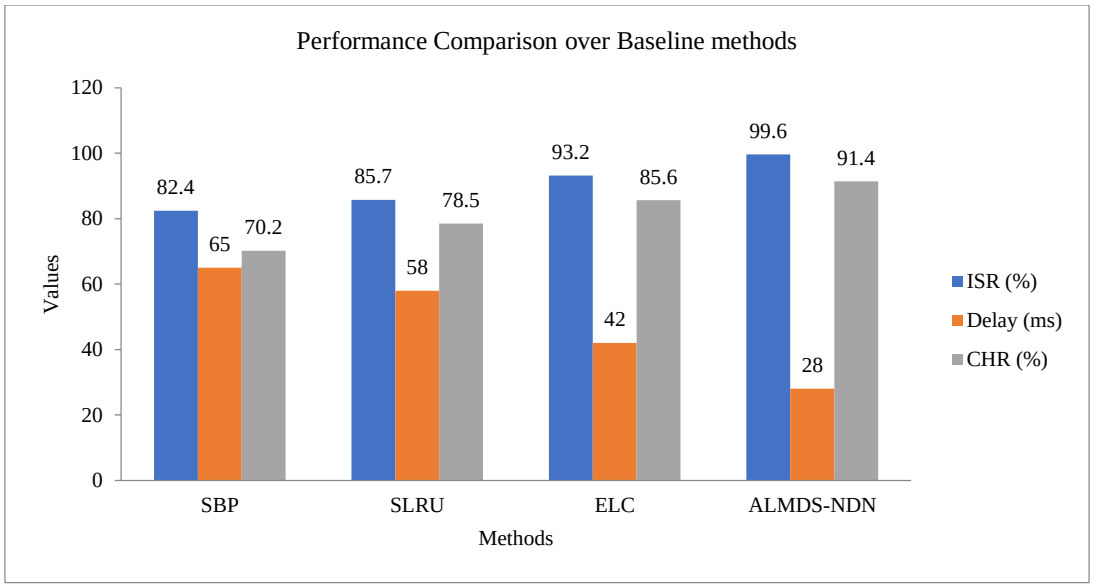


Fig. 7 Overall performance comparison with baseline methods

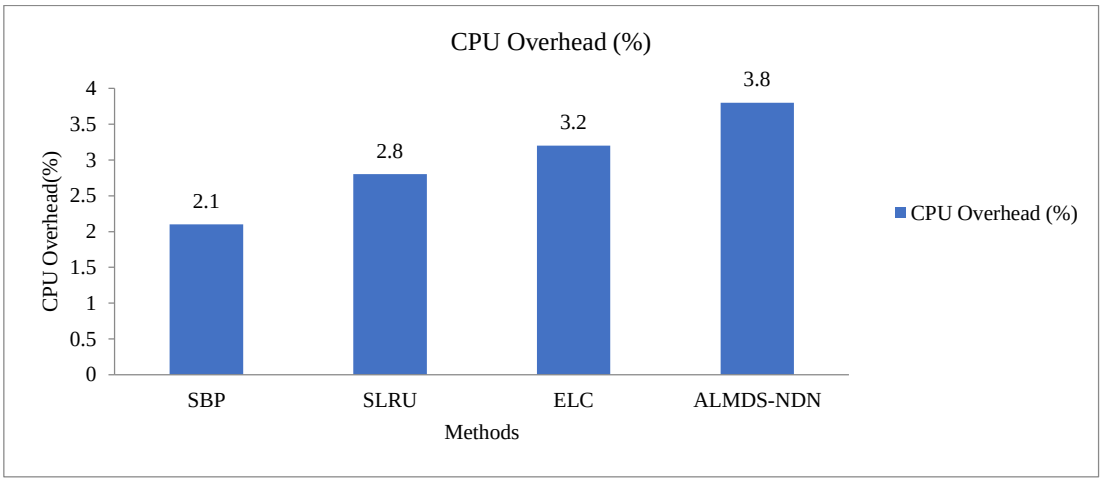


Fig. 8 Overall performance comparison with baseline methods

This is the main feature that indicates ALMDS-NDN offers a good ratio in terms of performance and computational complexity. As one can see from the above discussion, ALMDS-NDN offers significant performance in terms of efficiency, speed, and reliability.

As all already know, this approach utilizes ARIMA for anomaly detection, Isolation Forest for classifying malicious traffic, ARAC mechanism to control the behavior of nodes and DAIP to isolate malicious nodes from the rest of the network.

In comparison to other approaches, ALMDS-NDN shows better results in terms of detection accuracy, cache utilization, reduction of latency, and resilience to large-scale attacks.

5.6. Ablation Study

In order to analyze the influence of each layer used within the ALMDS-NDN approach, an ablation study is conducted where each layer is selectively eliminated, and its effects on the performance of the framework are measured.

Table 2. Ablation study performance comparison

Configuration	ISR (%)	Delay (ms)	CHR (%)
Full Model (ALMDS-NDN)	**99.6**	**28**	**91.4**
Without ARIMA	94.8	40	85.2
Without Isolation Forest	92.3	45	83.6
Without ARAC	89.7	52	80.1
Without DAIP	91.5	48	82.4

It allows assessing the importance of the Monitoring (ARIMA), Detection (Isolation Forest), Mitigation (ARAC), and Response (DAIP) layers needed to maximize both efficiency and security of the developed system. To conduct an analysis, several configurations have been prepared that

include the following combinations of the selected methods: use of all four methods; absence of ARIMA layer where the process of traffic monitoring does not occur; absence of the Isolation Forest layer where the step of anomaly detection is skipped; lack of ARAC where adaptive reputation-based control is not utilized; and absence of DAIP where the system's response mechanism does not function. From Table 2, it is evident that the removal of each component negatively affects the efficiency of the ALMDS-NDN system, although the magnitude of the impact differs among the components. When the Monitoring Layer based on ARIMA is removed, the ISR level decreases from 99.6% to 94.8% (a drop of 4.8%), and the delay increases. Thus, early anomaly detection plays an important role in this case. Without the Isolation Forest, there will be noticeable changes in the detection accuracy, with ISR decreasing to 92.3% (a drop of 7.3%), which will affect the system under attack. In the absence of the ARAC algorithm, the performance drop is the largest, with ISR declining to 89.7% (a drop of 9.9%), as it is responsible for adaptive traffic control and congestion management. The absence of DAIP leads to lower system stability because malicious nodes cannot be isolated properly, resulting in an ISR of 91.5% (a drop of 8.1%). Based on these results, ARAC has the greatest influence on ISR performance, followed by DAIP, Isolation Forest, and ARIMA. This quantitative ranking demonstrates that while all components contribute to system robustness, the adaptive mitigation capability provided by ARAC is the most critical factor in maintaining a high ISR under attack conditions.

6. Conclusion and Future Work

In this paper, the authors have proposed the ALMDS-NDN framework, which is a novel adaptive lightweight multi-layer defense system for securing Named Data Networking from multiple concurrent attacks such as Interest flooding, cache pollution, and cache poisoning. The framework combines traffic modeling using the ARIMA model, anomaly detection based on isolation forest algorithm, Adaptive Reputation-based Access Control, and Dynamic Attack Isolation and Purging mechanism into a single multi-layer structure. The experimental evaluations show that the proposed solution provides an excellent Interest Satisfaction Ratio (ISR) of 99.6%, achieves a low average end-to-end delay of 28 ms, and enhances the cache hit ratio to 91.4%. Further evaluation of detection accuracy, queue stability, reputation behavior, cache behavior, and attack resilience shows that the proposed framework can efficiently detect malicious activities and reduce their impact on the system performance. However, there are still areas that need further investigation for the improvement of ALMDS-NDN. For instance, the combination with deep learning-based adaptive systems may help improve the accuracy of complex and changing attack detections. Furthermore, applying and testing this framework in actual scenarios through large NDN testbeds can help gain more knowledge about scalability and feasibility. It would also be ideal if there is an extension of this framework towards other domains, such as the Internet of Things (IoT) and vehicular networking environment. Federated or decentralized learning may also be incorporated into this framework for added privacy and collaboration benefits. Lastly, optimization of energy usage should also be considered for effective deployment.

Conflicts of interest

The authors declare no conflict of interest.

Author contributions

Mr. A. Abdul Faiz led the project's conceptualization and investigation, developing the software and methodology while taking primary responsibility for the original draft.

Dr. N.A. Sheelaselvakumari provided formal analysis and supervision, contributing to the methodology and the critical review and editing of the final manuscript. Both authors have read and approved the published version.

Acknowledgments

The authors would like to thank the faculty and technical staff of the Department of Computer Science at Sri Krishna Arts and Science College for their continuous support and infrastructure. Special appreciation is extended to my supervisor for her continuous encouragement of this research work.

References

[1] Bengt Ahlgren et al., "A Survey of Information-Centric Networking," *IEEE Communications Magazine*, vol. 50, no. 7, pp. 26-36, 2012. [CrossRef] [Google Scholar] [Publisher Link]

[2] Farhan Ahmed Karim et al., "Named Data Networking: A Survey on Routing Strategies," *IEEE Access*, vol. 10, pp. 90254-90270, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[3] Mohammad Rasool Momeni et al., "Towards a Privacy-Preserving and Secure Smart City: Recent Advances in User-Centric Applications," *IEEE Internet of Things Journal*, vol. 12, no. 22, pp. 46392-46416, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[4] Najam U. Saqib, and Sani Isnain, "A Survey on Mitigation of Cache Pollution Attacks in NDN," *Acta Technica Jaurinensis*, vol. 18, no. 2, pp. 93-101, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[5] Jingcheng Song et al., "Eppda: An Efficient Privacy-Preserving Data Aggregation Federated Learning Scheme," *IEEE Transactions on Network Science and Engineering*, vol. 10, no. 5, pp. 3047-3057, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[6] Lin Yao et al., "Detection of Cache Pollution Attack based on Ensemble Learning in ICN-based VANET," *IEEE Transactions on Dependable and Secure Computing*, vol. 20, no. 4, pp. 3287-3298, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[7] Yasar Khan et al., "A Latency-Aware and Resource-Efficient Content Caching Scheme for Content-Centric Networks," *IEEE Access*, vol. 13, pp. 139650-139664, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[8] Abdelhak Hidouri et al., "A Detection Mechanism for Cache Pollution Attack in Named Data Network Architecture," *International Conference on Advanced Information Networking and Applications*, Sydney, NSW, Australia, vol. 449, pp. 435-446, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[9] Noriaki Kamiyama, and Takuto Kudo, "Investigating Impact of Fake-Type Content Poisoning Attack on NDN," *Proceedings of the 18th International Conference on Availability, Reliability and Security*, pp. 1-10, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[10] S.V. Vijaya Karthik, and J. Arputha Vijaya Selvi, "NDN Content Poisoning Mitigation using Bird Swarm Optimization and Trust Value," *Intelligent Automation and Soft Computing*, vol. 36, no. 1, pp. 833-847, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[11] Junseok Lee, and Dohyung Kim, "Heartbeat: Effective Access to Off-Path Cached Content in NDN," *IEEE Transactions on Network Science and Engineering*, vol. 12, no. 4, pp. 2974-2988, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[12] Ahmed Benmoussa et al., "Interest Flooding Attacks in Named Data Networking: Survey of Existing Solutions, Open Issues, Requirements, and Future Directions," *ACM Computing Surveys*, vol. 55, no. 7, pp. 1-37, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[13] Zhiwei Xu, Xin Wang, and Yujun Zhang, "Towards Persistent Detection of DDoS Attacks in NDN: A Sketch-based Approach," *IEEE Transactions on Dependable and Secure Computing*, vol. 20, no. 4, pp. 3449-3465, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[14] Syed Sajid Ullah et al., "Mitigating Content Poisoning Attacks in Named Data Networking: A Survey of Recent Solutions, Limitations, Challenges and Future Research Directions," *Artificial Intelligence Review*, vol. 58, no. 2, pp. 1-41, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[15] Matta Krishna Kumari, and Nikhil Tripathi, "Detecting Interest Flooding Attacks in NDN: A Probability-based Event-Driven Approach," *Computers and Security*, vol. 148, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[16] Simeon Ogunbunmi et al., "Interest Flooding Attacks in Named Data Networking and Mitigations: Recent Advances and Challenges," *Future Internet*, vol. 17, no. 8, pp. 1-21, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[17] Lalitha Chinmayee Hurali MaheshKumar, Sangeetha Jayavelu, and Annapurna P. Patil, "INI-24: A Dataset for Combating Interest Flooding Attack in Named Data Networks using ISP-RocketFuel Mapping Engine," *Automatika*, vol. 66, no. 4, pp. 638-649, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[18] Jiani Fan et al., "Differentiated Security Architecture for Secure and Efficient Infotainment Data Communication in IoV Networks," *International Conference on Network and System Security*, Denarau Island, Fiji, pp. 283-304, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[19] Rimsha Asif et al., "Advancing Named Data Networking: A Comprehensive Survey on Naming and Packet Lookup Mechanisms," *Peerj Computer Science*, vol. 12, pp. 1-58, 2026. [CrossRef] [Google Scholar] [Publisher Link]

[20] Ayoub Alsarhan et al., "Multi-Layer AI Sensor System for Real-Time GPS Spoofing Detection and Encrypted UAS Control," *Sensors*, vol. 26, no. 3, pp. 1-29, 2026. [CrossRef] [Google Scholar] [Publisher Link]

[21] Padmanaban G, Tamilselvi P, and Kavya S, "An Adaptive Multi-Engine Cyber Defense with Intelligent Deception and Real-Time Threat Mitigation," *ITM Web of Conferences*, vol. 85, pp. 1-7, 2026. [CrossRef] [Google Scholar] [Publisher Link]

[22] Hamed Alimohammadi et al., "Toward a Multi-Layer Defence Framework for Securing Near-Real-Time Operations in Open RAN," *IEEE Open Journal of the Communications Society*, vol. 7, pp. 480-497, 2026. [CrossRef] [Google Scholar] [Publisher Link]

[23] Lincong Zhao et al., "Adaptive Multi-Layer Defense Mechanism for Trusted Federated Learning in Network Security Assessment," *Computers, Materials and Continua*, vol. 85, no. 3, pp. 5057-5071, 2025. [CrossRef] [Google Scholar] [Publisher Link]

[24] Kaleem Ullah et al., "An Intelligent Multi-Layered Machine Learning and Big Data Analytics Framework for Autonomous Cyber Threat Detection, Adaptive Defense Mechanisms, and Secure Cloud-Native Smart Electrical Power Grid Cyber-Physical Infrastructure," *Spectrum of Engineering Sciences*, vol. 4, no. 3, pp. 1775-1802, 2026. [Google Scholar]

[25] Riddhi Mirajkar et al., "Adaptive Attack-Resistant Algorithm and Security Orchestration Framework for Named-Data Networking," *International Journal of Computer Information Systems and Industrial Management Applications*, vol. 18, no. 1s, pp. 555-570, 2026. [Google Scholar] [Publisher Link]

[26] Ramakrishna Kosuri et al., "Designing a Lightweight, Multi Layer Security Framework for IoT Devices: Adaptive and Scalable Protection against Emerging Threats in Connected Environments," *Proceedings of the 1st International Conference on Research and Development in Information, Communication, and Computing Technologies*, Nagapattinam, India, pp. 212-218, 2026. [CrossRef] [Google Scholar] [Publisher Link]

[27] Rashid Mustafa et al., "A Secure and Energy-Efficient Cross-Layer Network Architecture for the Internet of Things," *Sensors*, vol. 25, no. 11, pp. 1-36, 2025. [CrossRef] [Google Scholar] [Publisher Link]