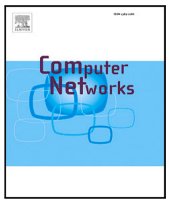
Computer Networks 271 (2025) 111604

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet



Deep Q-ICAN: A deep reinforcement learning-based approach for real-time CPA attack detection and mitigation in NDN architecture

Abdelhak Hidouri [a,b,g], Haifa Touati [a,*], Mohamed Hadded [c,d], Mohamed Amin Asri [c], Nasreddine Hajlaoui [e], Paul Muhlethaler [f], Samia Bouzefrane [g]

[a] Hatem Bettaher IReSCoMath Research Lab, University of Gabes, Gabes, Tunisia
[b] National School of Computer Science (ENSI), University of Manouba, Manouba, Tunisia
[c] Institute of Research and Technology (IRT SystemX), Paris, France
[d] Abu Dhabi University, Abu Dhabi, United Arab Emirates
[e] Unit of Scientific Research, Applied College, Qassim University, Unayzah, Saudi Arabia
[f] National Institute for Research in Digital Science and Technology (INRIA), Paris, France
[g] CEDRIC Lab, Conservatoire national des arts et métiers (Cnam), Paris, France

ARTICLE INFO

Keywords:
Future internet
NDN
Security
Intrusion detection
Cache Pollution Attack
Deep reinforcement learning

ABSTRACT

Named Data Networking (NDN) has emerged as a transformative architecture for next-generation Internet design, leveraging its data-centric paradigm to address scalability and security challenges inherent in traditional host-centric networks. By prioritizing content naming, stateful forwarding, and in-network caching, NDN inherently enhances efficient data dissemination and trust through cryptographic content validation. However, its reliance on distributed caching introduces vulnerabilities to Cache Pollution Attacks (CPA), where malicious actors inject illegitimate content to disrupt caching efficiency, degrade data availability, and inflate retrieval latency. Proactive mitigation of CPA is critical to preserving NDN's performance benefits and ensuring reliable content delivery, as unchecked attacks undermine consumer trust and network resilience. In this paper, we present Deep Q-ICAN: a deep reinforcement learning based intrusion detection and prevention system for NDN. It is an online, and adaptive approach designed to effectively mitigate CPA attacks and enhance the caching mechanism within the NDN architecture. Through extensive experiments in diverse and realistic topologies, we show that our solution achieves superior performance compared to previous approaches in the same field. In fact, Deep Q-ICAN achieves an accuracy of 98.87% and an Average Cache Hit Ratio of 80%. It improves the caching strategy by approximately 42% while maintaining the states of the NDN routers resources. Furthermore, Deep Q-ICAN significantly improves the Average Retrieval Delay (ARD) from 0.284s to 0.065s, thereby enhancing the efficiency of retrieving desired content from the Content Store (CS) for legitimate consumers.

1. Introduction

The evolving complexity of modern network architectures has amplified concerns around scalable security and efficient content delivery. Traditional TCP/IP infrastructures, with their host-centric design, face inherent vulnerabilities in addressing content privacy, routing scalability, and in-network caching efficiency. These limitations expose networks to sophisticated attacks such as Distributed Denial-of-Service (DDoS), and man-in-the-middle exploits, which exploit weaknesses in trust models and routing protocols [1]. Conventional security mechanisms, often reactive and reliant on centralized trust authorities, struggle to adapt to dynamic attack vectors and the growing demand for low-latency mitigation. Named Data Networking (NDN) emerges as a transformative architecture [2], leveraging name-based routing, decentralized trust via cryptographic signing, and stateful forwarding to inherently enhance security.

But despite the fact that NDN is a Secured-by-Design architecture, it still suffers from multiple types of attack that can impact its performance. One of those attacks is the Cache Pollution Attack (CPA). This attack not only causes network overload but also disrupts the timely delivery of legitimate content to consumers, leading to substantial performance degradation. The need for an emerging solution to detect and mitigate such an attack has always been an urgent need to ensure the well performance of the network. While there has been significant research on detecting and mitigating CPA, current approaches

* Corresponding author.
E-mail addresses: haifa.touati@cristal.rnu.tn, haifa.touati@univgb.tn (H. Touati).

https://doi.org/10.1016/j.comnet.2025.111604
Received 21 September 2024; Received in revised form 22 July 2025; Accepted 3 August 2025
Available online 26 August 2025

remain limited in their ability to adapt to the dynamic and evolving nature of NDN networks. In particular, existing methods often fail to balance the need for accurate threat detection with the scalability required for large-scale NDN deployments. This research proposes a novel solution based on Deep Q-learning, a type of Reinforcement Learning (RL), to address these challenges. Deep Q-learning, is highly suited for dynamic environments like NDN due to its ability to learn complex patterns in network traffic and adapt its defenses in real-time, effectively countering evolving CPA threats. By using powerful neural networks, Deep Q-learning can analyze intricate network behaviors, allowing for sophisticated and adaptive CPA detection and mitigation strategies. The state–action-reward structure of Q-learning naturally aligns with the CPA detection problem, where states represent network conditions, actions correspond to defensive measures, and rewards indicate the success of these measures. This alignment allows for the creation of highly efficient and effective CPA mitigation protocols tailored specifically to NDN architectures. Additionally, Q-learning's ability to balance exploration and exploitation enables the agent to continuously refine its detection and mitigation strategies, which is essential for maintaining an effective defense against evolving CPA threats. Furthermore, the scalability of deep Q-learning makes it particularly suitable for large-scale NDN networks, where rapid, accurate CPA detection and mitigation are critical. This work builds on our previous RL-based solutions [3,4], addressing the limitations identified in [3], where we introduced a Q-learning-based solution. While [3] demonstrated high accuracy, it was constrained by its reliance on tabular Q-learning, which faced scalability challenges due to state space explosion in large-scale networks. This necessitated the use of discrete state representation mechanisms to artificially constrain the problem dimensionality. In [4], we explored the potential of Deep Reinforcement Learning (DRL) to mitigate these issues. This journal article extends [4] by presenting Deep Q-ICAN, a fully validated Deep Q-learning framework that incorporates rigorous hyperparameter optimization, comprehensive evaluation under diverse conditions, and enhanced resilience to dynamic attack scenarios. These advancements ensure superior robustness, and adaptability in NDN network environments compared to prior work. The primary research gap addressed in this paper is the application of Deep Q-learning to CPA detection and mitigation within the dynamic NDN architecture, which has not been thoroughly explored in existing literature. Our contributions are as follows:

1. We introduce Deep Q-ICAN, a novel Deep Q-learning based Cache Pollution Attack Detection and Mitigation in NDN. This approach involves real-time data collection, detection, and mitigation via a Deep Q-Learning agent placed atop the NDN routers.
2. We introduce a pre-impact detection system to identify CPA attacks. Unlike many state-of-the-art post-impact solutions, which often rely on detecting attacks after they occur — typically by targeting NDN routers or storing malicious content — Deep Q-ICAN proactively identifies and mitigates CPA before any harm can be done to the Content Store (CS) caching process. Deep Q-ICAN discards the malicious interest rather than the already stored data content. Furthermore, it should be emphasized that this approach represents a novel way to manipulate the interest packet instead of the data packet itself.
3. We conduct extensive experiments to demonstrate the effectiveness of Deep Q-ICAN in a well-established simulator, i.e. NDNSim. The results demonstrate its superiority compared to recently published state-of-the art methods in various complex and realistic topologies. In terms of detection and mitigation effectiveness, our agent is capable of safeguarding the NDN router while maintaining one of the highest levels of CHR and ARD. Specifically, during a CPA attack, Deep Q-ICAN increases the CHR from 0% to 98%, and decreases the ARD from 0.284 s to 0.065 s. Furthermore, in terms of the proficiency of the agent, Deep Q-ICAN demonstrates a remarkable level of accuracy, outperforming state-of-the-art mechanisms with an accuracy rate of 98.87%.

The remainder of this paper is structured as follows: In Section 2, we present the underlying principles of the NDN architecture and provide a general explanation of the impact of the CPA attack on this architecture. Section 3 provides a concise overview of current literature, showcasing advanced work in the field. Section 4 introduces our proposed approach, Deep Q-ICAN. Section 5 provides an in-depth exploration of Deep Q-ICAN simulation process. Result analysis and discussion are covered in Section 6, while Section 7 evaluates our scheme by comparing it with other approaches. Section 8 presents an in-depth discussion about the limits and trade-off of our novel mechanism. Finally, Section 9 concludes our work and provides insights into future directions.

2. Background

In this section, we provide an overview of the NDN architecture and its associated security threats, with particular focus on the CPA attack, as detailed below.

2.1. NDN overview

In 2009, VAN Jacobson presented the initial proposal for NDN [5]. This concept later evolved into NDN as part of the NSF-funded future Internet architecture project. NDN employs two specific packets, called interest packets and data packets. The interest packet is always initiated by the consumer to request specific data, while the data packet contains the content as a response to the interest packet [6,7]. NDN is a clean-slate design that prioritizes content over data location [8], thus enhancing the likelihood of retrieving requested data from the optimal source in the network. Additionally, NDN eliminates the need for multiple network services, such as Network Address Translation (NAT) and Domain Name System (DNS), as the data location and IP addresses are no longer necessary. In NDN, each node comprises three essential data structures: the CS, the Pending Interest Table (PIT), and the Forwarding Information Base (FIB). The CS is designed to cache data packets, enabling the service of consumers without requiring continuous interest requests to the producer. This caching mechanism effectively reduces bandwidth usage and enhances content retrieval efficiency. The PIT temporarily stores unsatisfied interest requests and their corresponding interfaces until they are fulfilled. The FIB, on the other hand, is responsible for forwarding interest packets to the appropriate next hops. Unlike traditional IP-based routing, NDN's FIB entries are categorized by name prefixes rather than IP addresses. These entries include information about the next hops, allowing routers to efficiently forward interest packets to one or multiple next hops according to the forwarding strategy. This feature enables efficient multipath forwarding within the network [9,10].

2.2. CPA attacks in NDN

Despite possessing numerous characteristics, NDN is highly susceptible to various attacks, including CPA, Content Poisoning Attacks, Interest Flooding Attack [11], and Cache Privacy Attack. Among these, CPA stands out as particularly prominent within the NDN architecture [12]. CPA represents a form of cache process attack, where an attacker prevents legitimate consumers from obtaining the desired content from the cache of the CS. This attack intentionally depletes resources such as storage space and CPU, exhausts the router CS caching process, increases bandwidth usage, and induces delays in content retrieval [13]. The malicious consumer executes this attack by flooding the network with an excessive amount of malicious content, intending to store it in the CS. By overwhelming the system, the attacker depletes

NDN resources, rendering them inaccessible to legitimate consumers. This disruption affects network operations, denying honest consumers access to content cached in the neighboring CS of the NDN router [14].

There are two main types of CPA: Locality-Disruption Attack (LDA) and False-Locality Attack (FLA).

LDA reduces cache efficiency by manipulating the locality of the data. Attackers achieve this by continuously requesting unpopular, novel contents, thereby disrupting the natural patterns in consumer requests. This manipulation pollutes the cache, lowers the CHR for legitimate consumers, and ultimately harms their performance. For instance, while users might frequently request "cnn/news/meteo/video1.mp4", an attacker floods the cache with requests for unrelated malicious content like "hacker/malwares/malware01.bat", "hacker/malwares/malware02.bat", etc. This significantly increases the miss ratio and ARD, hindering legitimate consumers' access to cached content [15,16].

FLA, on the other hand, aims to decrease the hit ratio for regular consumers by repeatedly requesting the same set of contents. This creates artificial locality within the cache of the CS. This attack is effective because the attacker can quickly re-populate the cache with their chosen content. For example, if an attacker continuously requests content with the prefix "hacker/malwares/malware01.bat", this malicious content can dominate the cache. Consequently, the cache predominantly serves this malicious content, increasing cache misses and decreasing cache hits for legitimate requests, thus degrading their experience [3,17].

3. Related work

In the field of NDN caching security, numerous studies have been conducted to apply various techniques for CPA detection and mitigation [18]. Recent research has focused on developing innovative approaches to address this critical vulnerability in NDN architecture and we present them as follows :

The authors in [19] propose a new cache replacement method based on the Radial Basis Function Neural Network (RBFNN) to detect and mitigate CPA in NDN. The RBFNN framework is constructed using input associated with the inherent characteristics of cached content and output data associated with the content type, i.e. locality disruption, false-locality, and healthy. However, this approach recognizes several limitations. It does not explore the potential impact of the suggested technique on the overall operation of the NDN network, such as latency or throughput. Additionally, the mechanism may exhibit a degree of greediness in terms of memory and CPU usages. Moreover, the mechanism is capable of producing accurate results based on labeled datasets, but it may encounter challenges when implemented in real-time scenarios within the NDN network due to multiple pattern changes.

The authors in [20] leverage Fuzzy Restricted Boltzmann Machine (FRBM), for detecting and mitigating network anomalies in NDN. The framework integrates a cache replacement algorithm, known as Relationship-Based Access Control (ReBac), that operates on a reward-based mechanism to address CPAs. As part of the methodology, an essential metric involves monitoring the count of interest packets, providing valuable insights into the nature of the consumer, whether they are malicious or legitimate. However, the proposed framework could require considerable computational resources, potentially presenting challenges in environments with limited resources.

The authors in [21] propose a CPA detection algorithm based on the Gradient Boost Decision Tree (GBDT). The algorithm gathers node status information and path information as features, then the GBDT combines these two features to construct a cache pollution detection model. However, the proposed algorithm is evaluated using a limited dataset, and its performance may exhibit variations when applied to a larger dataset. Additionally, the algorithm exclusively considers only two features for cache pollution detection, potentially overlooking other features that could enhance the detection accuracy.

Liu et al. [22] introduce a detection scheme, named BO-CatBoost, based on Bayesian optimization and CatBoost to identify CPA. The paper analyzes the impact of CPA attack by extracting network traffic features, including the number of CacheMisses and the number of PIT entries. However, the proposed detection scheme might demand a substantial amount of training data to achieve the target accuracy, which may not be feasible in certain scenarios. Additionally, the scheme may face challenges in handling complex attacks involving multiple patterns of CPA attack, such as in the case of LDA.

Lin Yao et al. [23], propose a scheme based on hybrid heterogeneous multi-classifier ensemble learning (HHMCE) to detect CPA in NDN. The methods used in this paper are: (1) Support Vector Machine (SVM), (2) Neural Network (NN), and (3) Clustering. Each cluster head selects a different machine learning method to generate its own classifier based on all the statistics in its cluster. Whenever the membership changes by half, the classifier needs to be retrained. The proposed scheme operates under the assumption that attackers aim to saturate the buffer space with non-popular contents through the release of fake requests. However, this assumption may not always hold in LDA attack scenarios. Furthermore, the proposed scheme necessitates a substantial amount of communication overhead among NDN routers for cluster establishment and maintenance. This requirement may pose challenges, especially in scenarios involving a large-scale number of NDN nodes. Additionally, the scheme relies on the assumption that all nodes are honest and cooperative, which might not align with deployment conditions and could introduce vulnerabilities in practical implementations.

The authors in [24] proposed a mitigation mechanism called ICAN, which is a real-time monitoring approach focused on the state of the Cache of the CS. This statistical technique defines a set of parameters governing the CPA detection process, including Average Cache Hit Ratio (AVG-CHR), Average Interest Inter-Arrival Time (AVG-IAT), Hop Count, and prefix variation. ICAN operates by monitoring these parameters and detecting CPA attacks when their values exceed preset thresholds, making it a threshold-based approach. However, ICAN still faces challenges, such as potential sensitivity to the selection of thresholds, which could impact detection performance under diverse attack scenarios.

Cao et al. [25] present an SVM-based solution designed to distinguish between legitimate and malicious content in NDN. The proposed approach involves monitoring consumer behavior and various network parameters, such as the count of interest packets, the number distribution of requesters, core network traffic, and CHR, for effective attack detection. This supervised learning-based approach, while effective in scenarios with well-defined labels, may struggle when dealing with novel or previously unseen attack patterns.

Zhuang Du et al. [26] introduce a game theory-based approach for detecting and mitigating CPA in NDN. Their method utilizes popularity analysis to identify CPA, logistic regression for precise estimation of network and attack states, and a punishment mechanism to deal with attackers. It is important to note that the mechanism requires adjustment for pre-defined thresholds. Furthermore, the solution warrants additional evaluation in scenarios where the attacking pattern may change over pre-defined time steps. Additionally, the mechanism exhibits a degree of greediness in terms of space storage consumption.

Babu et al. [27] introduced a novel mechanism called SEND-DL which can be summarized in several key methods elaborated as follows : (1) Feedback Packet Mechanism: where this packet made a new type of Interest packet called "Feedback". This packet is specifically designed for three main functions : Identification of Data Packets, Router Response to Feedback Packets, Dynamic Path Adjustment and Handling Unsuccessful Verification. (2) Convolutional Neural Networks (CNN) : This technique is used for CS clean-up, detecting and removing bogus data packets from the CS. (3) Deep Belief Networks (DBN) : The DBN predicts the probability of each class during inference, which aids in identifying malicious sources within the NDN network. However,

this mechanism suffers from multiple issues such as : This mechanism can achieve continues increased complexity when applying the two types of deep learning algorithms, which can exhaust the resources of the NDN routers. The proposed method could introduce additional overhead in terms of processing and memory usage. Also, the method can face multiple issues when it comes to dealing with multiple patterns of the attack. The authors also did not deal to test the efficiency of the mechanism in both the LDA and FLA attacks structures. In addition to that, the signature mechanism can also be used to distinguish the requesters and reveal their identity. The authors used a limited dataset to simulate their solution, where this can be not the valid under real attack conditions. The mitigation in this mechanism can face the issues of the post-impact mitigation, where in order to detect the attack, the NDN router needs to suffer from a couple of malicious content that need to be cached.

The authors in [3] integrate reinforcement learning agents within NDN routers to detect and prevent CPA threats. Q-ICAN utilizes metrics such as CHR variation, interest inter-arrival time (IAT), and HopCount to differentiate between malicious and legitimate interests. By leveraging Q-learning algorithms, the system learns to predict and mitigate CPA attacks in real-time, enhancing the security and efficiency of NDN networks. Simulation results demonstrate Q-ICAN's effectiveness, achieving high accuracy rates in CPA detection, maintaining high CHRs, and reducing Average Retrieval Delays (ARDs). Despite its effectiveness, Q-ICAN faces limitations such as the challenge of handling a large number of states which can lead to increased computational complexity. Additionally, there may be a delay in properly detecting attacks and mitigate it. In [4], we investigated the potential of DRL to detect CPA in NDN. Specifically, we proposed a preliminary implementation of a Deep Q-Network (DQN) to replace tabular Q-learning, thereby eliminating the need for state-space reduction, a process that was performed manually in [3]. In this work, we significantly advance our initial approach by developing a fully validated Deep Q-learning framework called Deep Q-ICAN, which incorporates (1) rigorous hyperparameter optimization to ensure robust performance across diverse network and attack scenarios, (2) enhanced attack pattern resilience through testing against a broader spectrum of sophisticated and evolving CPA strategies, and (3) a detailed performance analysis, including detection accuracy, mitigation effectiveness, resource utilization, and latency. These advancements transition Deep Q-ICAN from a proof-of-concept [4] to a rigorously validated system ready for the applicable NDN deployment.

Discussion:

As summarized in Table 1, the existing literature on CPA attack detection and mitigation in NDN networks presents several significant limitations. Many current solutions, including those proposed in [19,22,25,26], and [27] exhibit a fundamental constraint by focusing exclusively on FLA attacks, thus demonstrating reduced effectiveness against LDA attack variants. Moreover, most existing CPA detection mechanisms, such as [20,23,25,27], employ offline training methodologies and predefined datasets for model development. This approach presents two primary limitations: (1) temporal mismatches between training data and operational contexts due to network dynamics, and (2) vulnerability to evolving CPA attack patterns. Additionally, these solutions often experience optimization challenges, manifesting as overfitting or suboptimal local solutions. The fixed nature of hyperparameters and detection thresholds, established during offline training, further compromises the adaptability to emerging attack patterns. Finally, the mitigation phase of existing solutions introduces additional constraints. For example, current approaches, such as [19,23,25,27], typically implement post-storage removal (Post-Impact) of malicious content from the CS, resulting in three main drawbacks: (1) degradation of caching efficiency, (2) reduction in ARD, and (3) unnecessary resource exhaustion of NDN routers. In contrast, interest-based (Pre-Impact) mitigation strategies demonstrate superior efficacy by preserving the integrity of the CS. This proactive approach enables effective CPA defense without compromising NDN's core caching functionality.

In summary, existing studies in the literature exhibit several limitations, including a lack of adaptability to the dynamic NDN environment, challenges in handling multiple CPA attack scenarios (e.g., LDA and FLA), and reliance on static training datasets. These approaches often struggle to address evolving attack patterns and fail to optimize mitigation strategies effectively. In contrast, our DRL-based approach overcomes these limitations by offering adaptability to dynamic network conditions, continuous learning capabilities, and proactive mitigation strategies.

4. Deep Q-ICAN proposed mechanism

4.1. Deep Q-ICAN prediction strategy

The primary motivation behind the development of our mechanism based on Deep Q-Learning was to address environments that involve continuous action and states, such as the NDN network. The conventional Q-Learning algorithm can effectively handle a small and discrete number of states and actions. However, in environments with a large state space, storing and updating a Q-table for every possible state–action pair can be memory-intensive. Moreover, if multiple variables need to be defined for each possible state in the environment, the Q-table becomes excessively large and impractical, since a larger Q-table, with more entries, significantly prolongs the time it takes for agents to explore every state and update the Q-Values in which it becomes increasingly unfeasible. In these cases, Deep Q-Learning, with its function approximation, is more efficient. In fact, Deep Q-Learning provides a more viable alternative, because it employs a deep neural network to approximate the Q-table. Deep Q-Learning replaces the conventional Q-table with a neural network. Instead of mapping a (state, action) pair to a Q-value, the neural network maps input states to (action, Q-value) pairs. This enables the algorithm to handle high-dimensional state spaces, which can be exceedingly large in a complex NDN environment.

To properly model our Deep Q-ICAN mechanism, it is essential to consider two fundamental phases present in every Deep Q-Learning mechanism: exploration and exploitation. These phases represent two aspects of decision-making. Specifically, the trade-off between exploration and exploitation entails a feedback loop between gathering information about the NDN environment and making decisions within that environment, as illustrated in Fig. 1. To effectively manage the balance between exploration and exploitation, our model incorporates the ϵ-greedy strategy. This widely-used pragmatic approach involves employing a fixed ratio of the exploration and exploitation phases (see Algorithm 1). The main idea is to choose the action with the highest policy value, while still allocating a portion of occurrences to explore a randomly selected alternative action. For example, if we assume that ϵ is equal to 0.1, then 90% of the time, the action deemed to be the best at that specific time step is chosen accordingly. The remaining 10% of the time, a different action is randomly selected [28].

Our model evaluates actions based on the outcomes they generate. The sole objective of the agent is to maximize the reward, aiming to discover action sequences that will help it achieve its objectives. For ease of description, the objective function that defines the goal of our agent is as follows:

$$\sum_{t=0}^{\infty} \gamma^t r(x(t), a(t)) \tag{1}$$

where x represents the state, a the action taken in that state, r the reward function, and γ a discount factor ranging between 0 and 1.

The input states are mapped to an (action, Q-value) pair by the main and target neural networks. In this scenario, each output node (which represents an action) holds the Q-value of the action as a floating point number. For ease of description, Deep Q-ICAN is trained in successive time steps, during which it performs the following operations:

Table 1
CPA detection and mitigation mechanisms and their drawbacks.

Ref.	Year	Used technique	Drawbacks	CPA attack type	CPA detection	CPA mitigation
[19]	2020	RBFNN approach	• It does not explore the potential impact on the latency or throughput. • Greedy in terms of memory and CPU usages. • Encounter challenges due to multiple pattern changes.	FLA	Offline detection	Data-based
[20]	2020	FRBM approach	• Greedy in terms of computational resources. • Can face challenges in environments with limited resources.	FLA/LDA	Offline detection	Data-based
[21]	2021	GBDT approach	• Evaluated using a small and limited dataset. • Limited in terms of features.	FLA/LDA	Offline detection	Data-based
[22]	2022	BO-CatBoost approach	• Demands huge amount of resource consumption. • Demands huge time of training in order to train the model. • Can't handle the dynamic changes in the attacking patterns.	FLA	Offline detection	Data-based
[23]	2022	HHCE learning approach	• Does not consider LDA attack scenarios. • Necessitates a substantial amount of communication overhead among NDN routers. • Issues dealing with large topologies. • Relies totally on the assumption that all nodes are honest and cooperative.	FLA/LDA	Offline detection	Data-based
[24]	2022	ICAN approach	• Threshold-based. • Can fall into high false positive and false negative.	FLA/LDA	Online detection	Interest-based
[25]	2023	SVM-based approach	• May struggle with continuous changing attack patterns.	FLA	Offline detection	Data-based
[26]	2023	Game theory-based approach	• Requires adjustment for pre-defined thresholds. • Not considering the dynamic of the attacking pattern. • Greedy in terms of space storage consumption.	FLA	Offline detection	Data-based
[3]	2023	Q-ICAN approach	• Limited in terms of the number of the state space.	FLA/LDA	Online detection	Interest-based
[27]	2024	SEND-DL approach	• Didn't consider all case scenarios in CPA attack. • Huge memory usage, space storage and high complexity. • Not considering the dynamic of the attacking pattern. • Limited dataset.	FLA	Offline detection	Data-based

1. First, we initiate the experience replay, which stores past experiences obtained by interacting with the environment.
2. Next, we randomly select a subset of these experiences in real-time, which serves as input data for both the main network and the target network.
3. The main network predicts the Q-value for each action based on the current state and action of each real-time data sample.
4. The target network predicts the best Q-value among all actions that may be taken from the next state of each real-time data sample.
5. Using the predicted Q-values from both networks and the observed reward of the real-time data, we calculate the loss and perform retro-propagation to update the weights of the main neural network. The target network remains unchanged.
6. After a specified number of time steps t, we copy the weights from the main network to the target network. This process enables the target network to obtain improved weights, allowing it to predict more accurate Q-values.

4.2. Essential components of Deep Q-ICAN

A fundamental objective of a reinforcement learning algorithm is to train the agent on appropriate actions to take across diverse scenarios. Through the training phase, the agent acquires knowledge about optimal actions. This premise guides the development of the essential components of Deep Q-ICAN, outlined as follows:

4.2.1. State space

The state information in deep reinforcement learning encapsulates the agent's perception of the environment and serves as input features of our neural network. In the proposed Deep Q-ICAN mechanism, we initiated the process by identifying pertinent parameters, guided by the outcomes of the CPA impact analysis that we conducted in [24,29]. The referenced study performed a thorough analysis of the impact on diverse metrics during transitions between attack and normal operation

Algorithm 1 The trade-off between the exploration and the exploitation

procedure DEEP Q-ICAN
  $Q(s,a) \leftarrow initialize(Q(s,a))$ ▷ Initialize Q-function Q(s,a)
  $Q_{target}(s,a) \leftarrow initialize(Q_{target}(s,a))$ ▷ Initialize target network $Q_{target}(s,a)$
  $\epsilon \leftarrow initialize(\epsilon)$ ▷ Initialize exploration rate ϵ
  $\alpha \leftarrow initialize(\alpha)$ ▷ Initialize learning rate α
  $\gamma \leftarrow initialize(\gamma)$ ▷ Initialize discount factor γ
  for episode = 1 to $max\_episodes$ **do**
    $s \leftarrow (AVG\_CHR_i, AVG\_IAT_i, HopCount_i)$
    for step = 1 to max_steps **do**
      if random() < ϵ **then** ▷ Choose action a using epsilon-greedy policy
        a ← RandUniform() ▷ Uniform distribution over actions
      else
        a ← max_a(Q(s,a)) ▷ a = 0 : Discard Interest i and a = 1 : Let Interest i pass
      end if
      Process(a,s',r) ▷ Take action a in environment, observe next state s' and reward r
      $y \leftarrow r + \gamma \times max_{a'}(Q_{target}(s', a'))$ ▷ Calculate TD-target
      $\theta \leftarrow \theta - \alpha \times (y - Q(s,a)) \times \delta_\theta Q(s,a)$ ▷ Update Q-network parameters θ
      if $steps \mod target\_update\_freq = 0$ **then**
        $Q_{target} \leftarrow$ load_states()
      end if ▷ Update target network periodically
      Update(ϵ) ▷ Decrement exploration rate
    end for
  end for
  Return $Q(s,a)$ ▷ Return learned Q-function Q(s,a)
end procedure

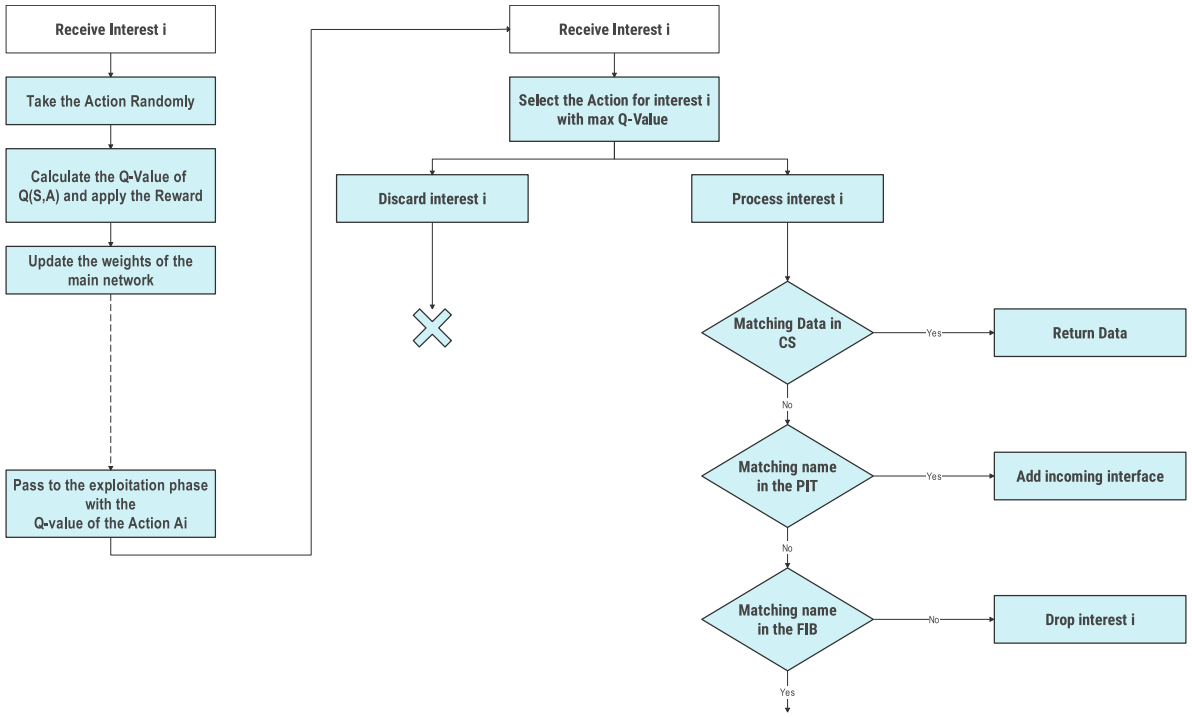


Fig. 1. The trade-off between the exploration and the exploitation.

scenarios. The identified metrics, subsequently utilized in constructing our state space components, are elaborated as follows:

Average Cache Hit Ratio (AVG-CHR) : The CHR is defined as the ratio of cache hits to total requests received.

$$CHR = \left(\frac{\sum \text{CacheHits}}{\sum \text{CacheHits} + \sum \text{CacheMisses}} \right) \times 100 \quad (2)$$

In our analysis of the parameters relevant to the detection of CPA, we investigated in [29] the CHR variation both in the presence and absence of CPA attacks. Our findings indicate a significant reduction in CHR during a CPA attack, dropping to 0% from its normal state. As a result, we suggest that the AVG-CHR might be a viable indicator for detecting the existence of a CPA attack.

Average Interest Inter-Arrival Time (AVG-IAT) : The IAT is defined as the time between two consecutive interest for the same data packet. Experiments that we have carried out in [24] have allowed us to draw the variation of the AVG-IAT in the presence and the absence of CPA attacks. We notice that when CPA attack occurs, the average IAT decreases significantly. Consequently, we can consider the average IAT as a suitable parameter to construct our state space.

Hop Count Variation : By definition, the hop count represents the number of hops that the interest travels between the consumer and the NDN router. In Deep Q-ICAN, the agent autonomously explores and learns optimal hop count values. This adaptive mechanism enables the system to dynamically adjust its decision-making process based on various stages of the CPA attack.

To sum up, the three metrics were chosen based on our prior work [3], which conducted an impact analysis of CPA attacks on NDN networks. The study revealed that CPA attacks significantly reduce the CHR due to malicious interest flooding. This makes AVG-CHR a reliable indicator of cache disruption caused by CPA attacks. Moreover, inspired by traditional DDoS detection methods in TCP/IP networks, the IAT between incoming packets is an effective metric for identifying anomalies in traffic patterns. In the context of NDN, CPA attacks typically generate a high volume of malicious interests in a short timeframe, resulting in a noticeable decrease in the AVG-IAT. Thus, this metric complements AVG-CHR by capturing temporal traffic anomalies.

Finally, the Hop Count Variation metric measures the consistency of paths taken by interest packets. Under normal conditions, the hop count for a specific prefix remains relatively stable. However, CPA attacks introduce significant path deviations as the network attempts to handle the high volume of malicious interests. By monitoring hop count variations, we can detect routing inconsistencies caused by CPA attacks.

It is worth noting that these metrics are complementary, collectively providing a robust detection mechanism by addressing various dimensions of the attack: AVG-CHR reflects cache-related anomalies, AVG-IAT captures temporal traffic irregularities, and Hop Count Variation identifies routing disruptions. Together, they complement each other, reducing the likelihood of detection errors. For instance, a false positive in AVG-CHR due to legitimate cache misses can be counteracted by stable AVG-IAT and hop count patterns, and vice versa.

4.2.2. Action space

The action space represents the opportunity given to the agent to achieve the expected objective. In this regard, a discrete space action should be optimal as much as possible. The reason behind this is to increase the performance of our agent throughout the NDN network while reducing the complexity of the training phase of the algorithm.

As shown in (2), once our router receives an interest, it is confronted with two possible actions, either to process or to discard such interest.

Accordingly, in our solution, it is crucial to observe that the rationale behind opting for the design of the action state is primarily centered around its interaction with the interests rather than the data. This choice is motivated by the objective of detecting the attack even before it has an impact on the cache of the CS, which is referred to as pre-impact in this context. This approach stands in contrast to the existing solutions discussed in the previous section, which rely on detecting the attack after it has already impacted the cache of the CS, referred to as post-impact. It is worth noting that the solution proposed by existing approaches, as discussed in Section 3, leads to a degradation in the overall performance of the CS cache. The overall model of our

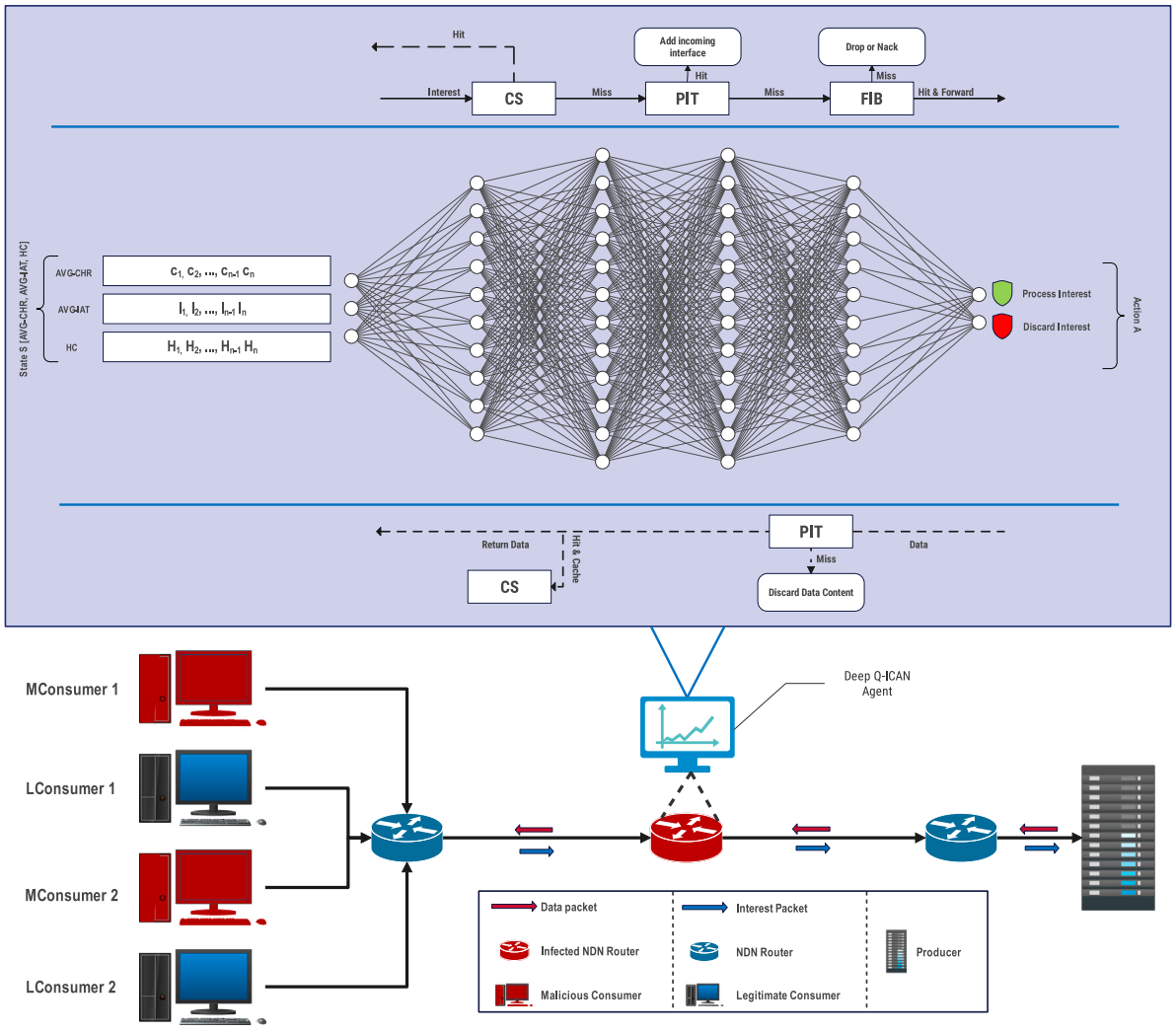


Fig. 2. Global View of the Deep Q-ICAN.

action space is designed as follows:

$$Action = \begin{cases} 1 & \text{Process interest} \\ 0 & \text{Discard interest} \end{cases} \tag{3}$$

4.2.3. Reward

The formulation of the reward function plays a crucial role in determining the effectiveness of the DRL algorithms. This aspect is particularly essential for facilitating the integration of DRL techniques into NDN networks, specifically within the context of the caching process in the CS of the NDN router. The reward function needs to accurately reflect the desired control behavior of our agent. It is important to highlight that the process of identifying a suitable reward function for this problem has proven to be quite challenging, mainly due to the unpredictable nature of the CPA attack. In order to properly model our rewarding function, we need to achieve several objectives, elaborated as follows:

- If the attack occurs and the decision is correct, AVG_CHR increases.
- If the attack occurs and the decision is wrong, AVG_CHR decreases.
- If there is no attack and the decision is correct, AVG_CHR increases.
- If there is no attack and the decision is wrong, AVG_CHR decreases.

With these results, we defined the reward using the following function:

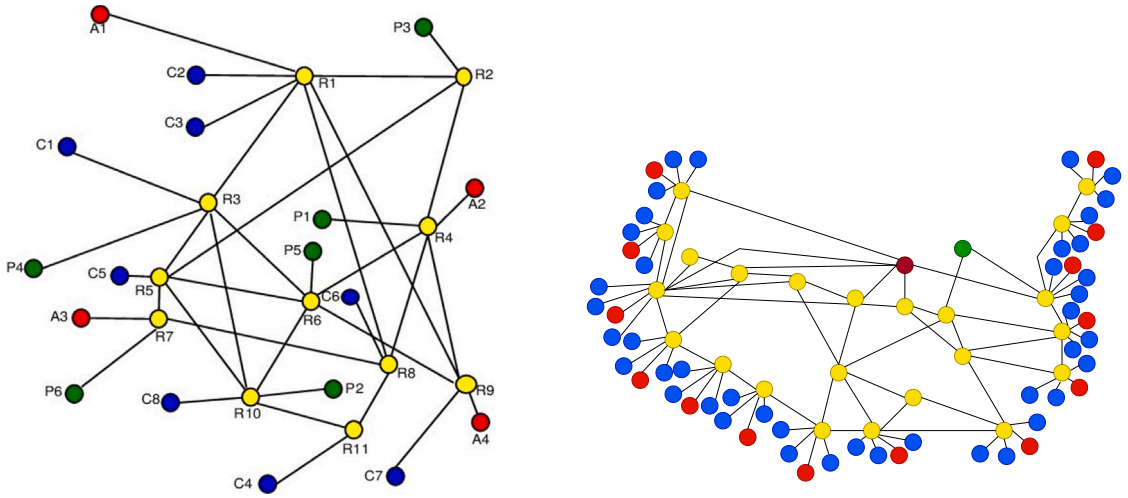
$$Reward = \{AVG\_CHR_{t+1} - AVG\_CHR_t\} \tag{4}$$

where

$$Step(t) = RTT_{\text{moy}} \tag{5}$$

RTT_{moy} helps our model assess overall network conditions more accurately than individual Round-Trip Time (RTT) measurements, which is a crucial metric to measure NDN network performance and latency. In addition, RTTmoy helps smooth out short-term variations in network latency, providing a more stable input for our model decision-making process, along with the ability to adapt its strategies based on current network conditions, potentially improving its performance across varying network states and affording suitable values for the reward function.

According to the definition above, when the agent makes incorrect decisions, the reward function yields a negative value, thereby serving as a penalty to the model. Conversely, in scenarios where the reward function yields a positive outcome, it will function as a form of positive reinforcement. In essence, our reward function shall grant rewards upon accurate decisions while simultaneously penalizing incorrect ones. Fig. 2 illustrates the global view of the Deep Q-ICAN mechanism.



(a) DFN topology. (b) AT&T topology.

Fig. 3. Simulation topologies.

5. Deep Q-ICAN simulation process

To evaluate the Deep Q-ICAN mechanism, the simulation has been made using the latest NDNSim module of NS3 (Network Simulator 3) [30–32]. To achieve our objectives, two main configurations need to be set, namely the "Attack Scenario Setup" and the "Deep Q-ICAN Agent Setup":

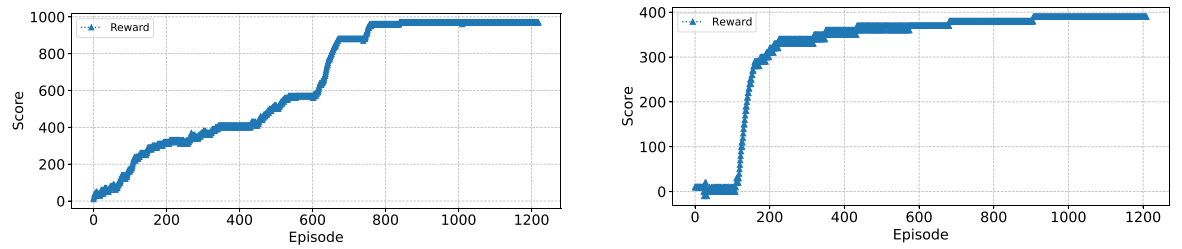
1. *Attack Scenario Setup:* To comprehensively evaluate the effectiveness of our mechanism, we considered two distinct topologies and designed elaborate attack scenarios. The Attack Scenario Setup is detailed in Table 2, and the topologies are depicted in Fig. 3.
    - *DFN topology:* Consisting of 29 nodes, with 4 nodes designated as malicious consumers executing CPA attacks. Experiments involve varying the number of attackers, ranging from 1 to 4, operating concurrently. Additionally, the attacker's interest rate is adjusted at values of 120, 160, 200, and 260. The attack onset times are varied with a 6-s separation. To present a more challenging scenario for our mechanism, we introduce variations in the cache size of the CS at 100, 150, and 200, coupled with changes in the cache replacement policy such as : Least Recently Used (LRU) and Least Frequently Used (LFU).
    - *AT&T topology:* This topology was a great challenge to our mechanism where the number of the nodes is extended to 80 compared to the DFN topology, 14 attacking nodes are simultaneously launched at the start of the simulation, with a 6-s separation between attacks. We also variated the attacker's interest rate (120,160,200 and 260), the NDN router CS size (100,150 and 200) and the caching policy (LRU and LFU). This intricate setup enables a comprehensive assessment of the robustness of our mechanism in a large-scale network environment.
2. *Deep Q-ICAN Agent Setup:* In addition to setting up the Attack Scenario, numerous parameter tuning experiments were conducted to establish optimal hyperparameters for the Deep Q-ICAN Agent. As detailed in Table 3, the hyperparameters are configured as follows: the training batches per time step are set to 1, the reward scaling is set to 10. During the pretraining phase, which encompasses 1222 steps, we defined the policy's hidden layer sizes as [128, 128, 256, 256], aligning them with the sizes of the Q-network's hidden layers, while the activation function policy is set to Rectified Linear Unit (ReLU). Moreover, to ensure the effective learning and performance of the agent, key reinforcement learning parameters were fine-tuned. ϵ is set to 0.99, the Learning Rate α is assigned a value of 0.1, and the Discount Factor γ is equal to 0.9.
As demonstrated in our previous work [4], we ensured that the hyperparameters chosen for our agent provide optimal performance. This was achieved by systematically varying the values of key hyperparameters, including the exploration rate (ϵ), the learning rate (α), and the discount factor (γ), while evaluating their impact on accuracy and the AVG-CHR. Furthermore, this selection process was later validated using additional verification steps, such as the Grid Search algorithm [33]. These steps ensured that the agent avoids both overfitting and convergence to suboptimal local solutions. In our simulation framework, an episode refers to a single interaction cycle between the Deep Q-Learning agent and the NDN environment. Each episode involves the agent observing the current network state, selecting an action, and receiving feedback in the form of a reward or penalty. This feedback loop allows the agent to iteratively learn and refine its policy. Over the course of the simulation, which spans 106 s, the system generates 1222 episodes. This corresponds to an average episode duration of approximately 87 ms, providing the granularity required to capture the dynamic nature of NDN traffic and the real-time adaptation capabilities of the Deep Q-ICAN agent.

6. Results analysis and discussion

In this section, we comprehensively evaluate the performance of our Deep Q-ICAN solution, employing a set of diverse metrics. To provide insights into the agent's capabilities in mitigating CPA and ensuring timely and accurate responses, we consider the following metrics: reward accumulation represented by the Score values, CHR, ARD, and agent efficiency metrics, including Accuracy (ACC), Recall, Precision, and F1 score.

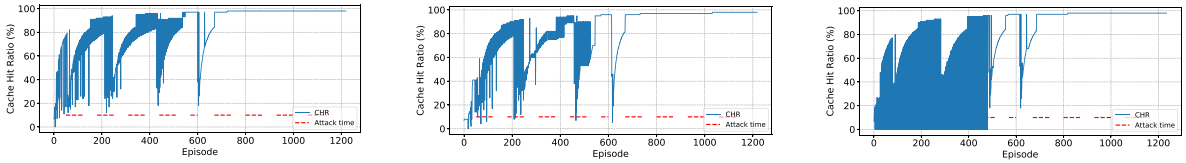
6.1. Evaluation of the score values

The score values define how well the algorithm enhances its results over time. Additionally, the scoring determines the behavior of Deep Q-ICAN under certain pre-defined scenarios. In our case, the score values were obtained under a CPA attack, as shown in Fig. 4. In the



(a) DFN topology. (b) AT&T topology.

Fig. 4. Deep Q-ICAN score under DFN and AT&T topologies.



(a) CS = 100. (b) CS = 150. (c) CS = 200.

Fig. 5. Performance of CHR while varying the CS cache size in the DFN topology.

Table 2
Deep Q-ICAN simulation settings.

Simulation parameters	Associated values
Simulation time	106 s
Number of nodes	**DFN :** 29/**AT&T :** 80
Number of legitimate consumers	**DFN :** 8/**AT&T :** 42
Number of attackers	**DFN :** 1, 2, 3, 4/**AT&T :** 14
Consumer type	ConsumerZipfMandelbrot
Attacker's interest rate	120, 160, 200, 260
Time period of the attack	5-12, 18-25, 31-38, 44-51, 57-60, 67-74, 80-87, 93-100 s
Router CS size	100, 150, 200
Router policy	LRU, LFU

Table 3
Deep Q-ICAN hyper-parameter.

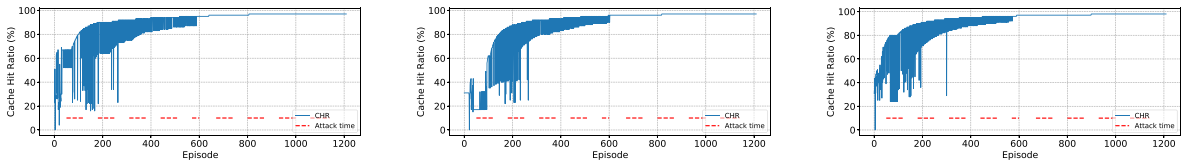
Deep Q-ICAN Hyper-parameters	Value
Training batches per timestep	1
Exploration noise	None (Stochastic policy)
Reward scaling	10
Number of pretraining steps	1222
Policy hidden sizes	[128,128,256,256]
Policy hidden activation	ReLU
Q hidden sizes	[128,128,256,256]
Q hidden activation	ReLU
Q weight decay	0
Learning rate	0.1
Discretization level	Default
Strategy of exploration and exploitation	Epsilon-greedy strategy
Epsilon (ϵ)	0.99
Discount factor (γ)	0.9

DFN topology, four attackers launch a CPA attack with 260 malicious interests/second, while in the AT&T topology, 14 attackers launch 260 malicious interests/second. The results show that in the DFN topology, the score stabilizes at around 780 episodes, while in the AT&T topology, stabilization occurs at 600 episodes. The simulation demonstrates that at around half of the total episodes, our mechanism is capable of learning and performing the right actions against CPA attacks.

6.2. Evaluation of the Cache Hit Ratio (CHR)

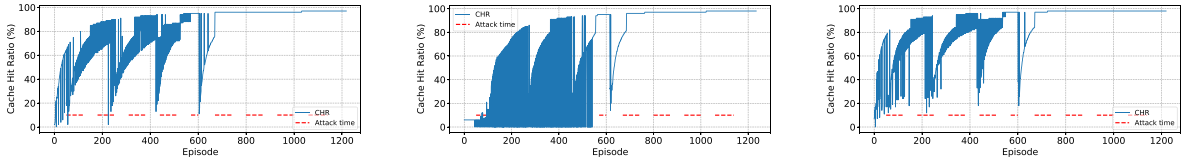
In order to evaluate the effect of our solution on the CHR, we set our mechanism in multiple case scenarios by varying: (1) the CS cache size, (2) the frequency of sending malicious interests, (3) the number of attackers, and (4) the caching policy. The results are detailed below:

- *Varying the CS cache size*: Varying the cache size serves as one of our evaluation metrics for assessing the effectiveness of our mitigation mechanism. The primary objective of this variation is to understand how our solution impacts the CS cache when a CPA attack occurs. In the DFN topology (refer to Fig. 5), we experimented with three main cache sizes (100, 150, and 200). The results consistently show that regardless of the cache size variation, our solution effectively detects and mitigates the CPA attack at 680 episodes. This accuracy is similarly observed in the AT&T topology (refer to Fig. 6), where stabilization occurs at nearly 600 episodes.
- *Varying the frequency of sending malicious Interests:* In order to effectively address various scenarios that could impact the cache of the CS, the variation of the frequency of sending malicious content is crucial. This variation can significantly degrade the performance of serving desired content, as explained in [3]. To assess the effectiveness of our solution, we varied the frequency of sending malicious interests per second in both DFN and AT&T topologies. The variation, illustrated in Figs. 7 and 8, encompassed multiple frequencies, including 120, 160, 200, 220, and 260 malicious interests per second. Across all defined frequencies, the CHR achieved a success rate of 98% in the DFN topology and 96% in the AT&T topology.
- *Varying the number of the attackers:* To significantly impact the effectiveness of CS performance, a malicious user may launch multiple attackers within pre-defined time steps. We assessed the effectiveness of our mechanism under various attacker scenarios. In the DFN topology, we evaluated scenarios with 1, 2, 3, and 4 attackers operating simultaneously. The results, illustrated in Fig. 9-a, -b, -c, and -d, demonstrate that our mechanism effectively counters the impact of these attackers. It learns to recognize



(a) CS = 100. (b) CS = 150. (c) CS = 200.

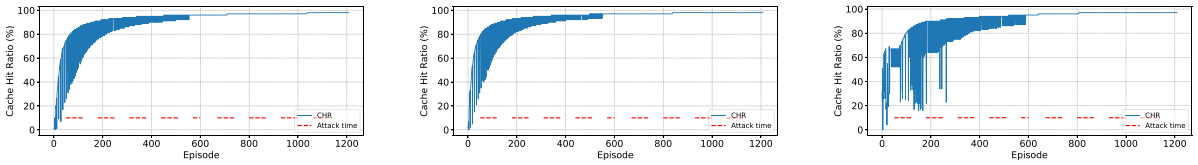
Fig. 6. Performance of CHR while varying the CS cache size in the AT&T topology.



(a) 120 interest/second (b) 160 interest/second (c) 200 interest/second

(d) 220 interest/second (e) 260 interest/second

Fig. 7. Performance of CHR while varying the frequency of sending malicious interests in the DFN topology.



(a) 120 interest/second (b) 160 interest/second (c) 200 interest/second

(d) 220 interest/second (e) 260 interest/second

Fig. 8. Performance of CHR while varying the frequency of malicious interests in the AT&T topology.

attack patterns within 550 to 700 episodes, subsequently boosting the CHR to approximately 98%.

- *Varying the caching policy:* NDN architecture introduces several pre-defined caching policies, including LRU and LFU. This variety of caching policies offers the CS multiple possibilities for caching from different perspectives, depending on the main desired usage. To evaluate the effectiveness of our mechanism in this diverse set of policies, we applied our mechanism in the DFN topology. In both LRU and LFU policies, our mechanism learned about the CPA attack and effectively mitigated it at episode 650. The CHR values increased from 0% to 98%, as illustrated in Fig. 10. The same level of effectiveness was observed when evaluating our mechanism in the AT&T topology where the CHR values increase from 0% to 96%, as shown in Fig. 11.

6.3. Evaluation of the Average Retrieval Delay (ARD)

The CPA is a type of attack that not only targets the cache of the CS, but also poses a significant threat to the delivery of content to consumers. In other words, CPA leads to an increase in the ARD of legitimate contents. Through simulations on the DFN topology, our mechanism demonstrates its ability to effectively reduce the ARD from 0.284 s to 0.065 s, establishing it as an essential means to deliver desired legitimate content and improve delivery time for consumers, as shown in Fig. 12. Furthermore, our mechanism outperforms our previous solution by a notable difference of 0.1357 s, demonstrating the superior performance of our novel solution compared to our previously designed mechanisms.

6.4. Evaluation of the agent's efficiency

To validate the efficiency of the Deep Q-ICAN agent, we consider in this subsection several metrics commonly used in deep learning evaluation, such as (1) Accuracy (ACC), (2) Recall, (3) Precision, and (4) F1 Score (F1).

As shown in Tables 4 and 5, the comparative analysis demonstrates significant topology-dependent behavior in Deep Q-ICAN's performance profile.

In the DFN topology, Deep Q-ICAN achieves a 3.78% improvement in accuracy (98.87% vs. 95.09%), indicating better cache efficiency. However, this comes at the cost of a 25.59% drop in precision (70.87%

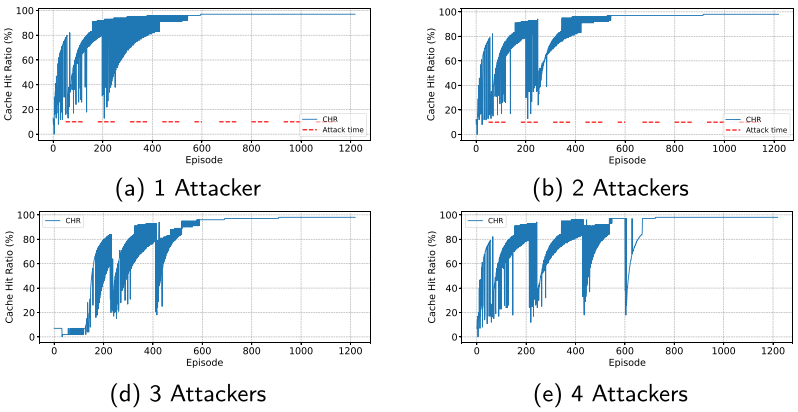


Fig. 9. Performance of CHR while varying the number of the attackers in the DFN topology.

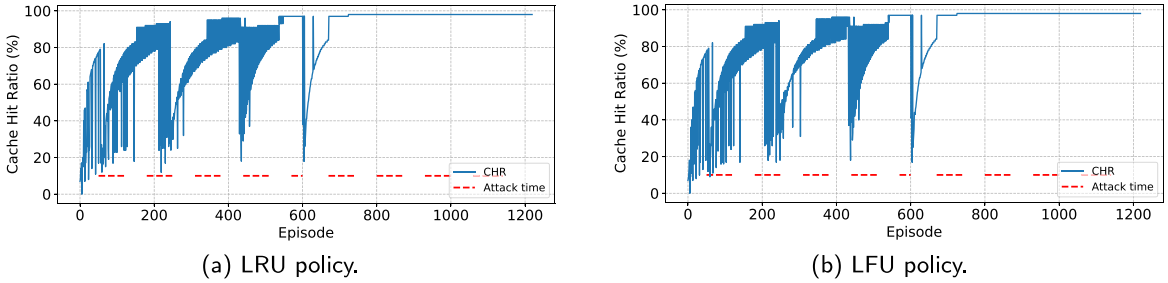


Fig. 10. Performance of CHR using different caching policies in the DFN topology.

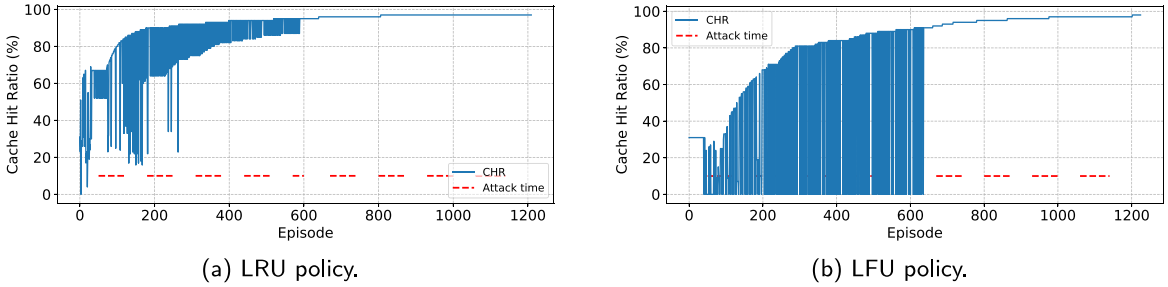


Fig. 11. Performance of CHR using different caching policies in the AT&T topology.

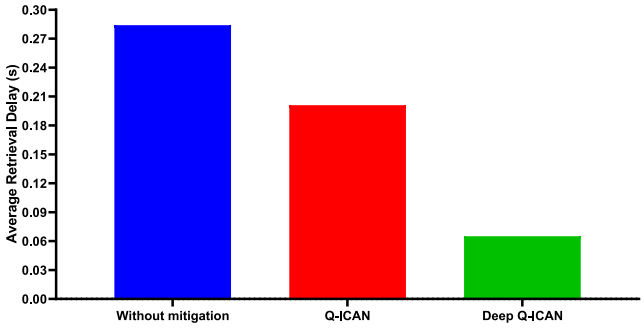


Fig. 12. Average retrieval delay under CPA.

vs. 96.46%) and a 16.90% drop in F1 score (80.29% vs. 97.19%). These trade-offs result from a deliberate design choice that favors comprehensive threat detection — maintaining a high recall (92.59% vs. 97.92%) - while tolerating more false positives to prevent CPA attacks. Furthermore, Deep Q-ICAN was explicitly configured to favor exploration during training, using a high exploration rate ($\epsilon = 0.99$) in the ϵ-greedy strategy. This means that the agent chose exploratory (non-greedy) actions 99% of the time, significantly increasing its chances of detecting rare or subtle threats [34]. While this configuration supports high recall, it also raises the likelihood of misclassifying benign traffic as malicious, explaining the observed drops in precision and F1 score — particularly in the simpler DFN topology.

In contrast, in the more complex AT&T topology, Deep Q-ICAN exhibits much better balance. The F1 score gap narrows to 3.12% (92.85% vs. 95.97%), and recall even surpasses Q-ICAN (99.63% vs. 97.65%). The precision reduction is limited to 7.40% (86.93% vs. 94.33%), and accuracy also improves slightly by 1.61% (96.62% vs. 95.01%). These results confirm Deep Q-ICAN's ability to scale effectively and maintain both efficiency and security in large, dynamic environments.

This behavior aligns with findings in the reinforcement learning literature [35–37]. Q-ICAN, based on tabular Q-learning, operates with fixed state–action mappings, yielding predictable and stable performance — especially effective in simple or deterministic topologies like DFN. In contrast, Deep Q-ICAN, built on Deep Q-learning, uses neural networks to approximate Q-values, enabling greater adaptability in complex scenarios like AT&T. This adaptability comes with known trade-offs in the literature: Deep Q-learning is prone to overestimation bias and aggressive exploration behavior, as documented

Table 4
Comparison between Deep Q-ICAN and Q-ICAN in the DFN topology.

Metric	Q-ICAN	Deep Q-ICAN	Δ
Accuracy (%)	95.09	98.87	+3.78
Recall (%)	97.92	92.59	−5.33
Precision (%)	96.46	70.87	−25.59
F1 Score (%)	97.19	80.29	−16.90

Table 5
Comparison between Deep Q-ICAN and Q-ICAN in the AT&T topology.

Metric	Q-ICAN	Deep Q-ICAN	Δ
Accuracy (%)	95.01	96.62	+1.61
Recall (%)	97.65	99.63	+1.98
Precision (%)	94.33	86.93	−7.40
F1 Score (%)	95.97	92.85	−3.12

Table 6
Comparative analysis of Deep Q-ICAN with ICAN and Q-ICAN.

	AVG-CHR	Accuracy	Memory usage
ICAN [24]	50.53%	92.3%	135.849 MiB
Q-ICAN [3]	72.55%	95.09%	160.509 MiB
Deep Q-ICAN	80%	98.87%	365,749 MiB

by Hasselt et al. [35]. In simpler settings (e.g., DFN), this can cause benign instances to be misclassified as threats, reducing precision and slightly lowering recall. Conversely, in complex environments (e.g., AT&T), such exploration results in more comprehensive threat coverage, achieving higher recall with only a minor drop in precision. In summary, Deep Q-ICAN is better suited for large-scale or complex network environments, while Q-ICAN remains advantageous in smaller or more structured topologies.

7. Qualitative Comparison Analysis (QCA)

In this section, we employ Qualitative Comparative Analysis (QCA) to assess the effectiveness of our solution in comparison to both our previous approaches [3,24] and the state-of-the-art solutions referenced. The subsequent paragraphs elucidate the details of these QCAs.

7.1. QCA between ICAN, Q-ICAN and Deep Q-ICAN

As demonstrated in the previous section, Deep Q-ICAN effectively detects and mitigates CPA attacks with high accuracy. This led us to conduct a QCA between Deep Q-ICAN and our previous solutions (ICAN [24] and Q-ICAN [3]).

As shown in Table 6, Deep Q-ICAN outperforms our previous solutions in terms of accuracy by almost 6% compared to ICAN and by 3% compared to Q-ICAN. Regarding cache performance, ICAN achieved an AVG-CHR of 50.53%, which was improved to 72.55% by Q-ICAN. Deep Q-ICAN further raises this metric to 80%, reflecting its superior ability to maintain cache efficiency under attack conditions. In terms of resource usage, Deep Q-ICAN consumes approximately 365.749 MiB of memory, which is higher than both Q-ICAN and ICAN. This increased memory consumption is primarily due to the use of a deep neural network (DNN), which requires additional memory to store the weights and biases of each neuron. In contrast, Q-ICAN relies on Q-tables, where each state–action pair stores a single Q-value. Q-ICAN consumes approximately 160.509 MiB, while ICAN uses only 135.849 MiB, as it employs lightweight statistical measures for detection. Deep Q-ICAN's memory footprint is further influenced by other factors, including the size of the state and action spaces, the replay buffer, and interactions with the NDN router, factors known to increase memory usage, as reported in prior work [38,39].

7.2. QCA between Deep Q-ICAN and state-of-the-art solutions

To thoroughly evaluate the efficiency of Deep Q-ICAN, we conducted a comprehensive QCA between our mechanism and the previously cited state-of-the-art solutions.

Firstly, Deep Q-ICAN is represented in the NDN network as an independent agent located within the NDN router. More precisely, it autonomously collects its states without the need for any pre-defined datasets, which could potentially lead to overfitting in terms of the obtained results. Our real-time interactive agent achieves near-optimal performance across diverse operational scenarios, as empirically validated in Section 6. While it outperforms contemporary state-of-the-art solutions in several aspects, particularly in scalability and threat coverage, it does so with trade-offs in certain precision metrics, especially in simpler environments.

Secondly, Deep Q-ICAN sustains high detection effectiveness and cache efficiency, achieving an accuracy of 98.87% and an AVG-CHR of 80%, which is significantly higher compared to most of the presented state-of-the-art solutions.

Thirdly, our intelligent mitigation mechanism is a pre-detection and defense solution (pre-impact), which sets it apart from other existing solutions, making it unique in this perspective. This distinction is largely attributed to the way it takes action. More in depth, Deep Q-ICAN relies on discarding the suspected malicious interest or letting pass the legitimate interest packet based on the optimal associated Q-value predicted in this case. This represents a huge advantage, where the other state-of-the-art solutions need to receive the first couple of damages from the malicious contents in order to detect the presence of the attack (post-impact). A huge stressful scenario is set to test our solution, specifically the usage of realistic topologies such as the DFN and AT&T topologies.

Moreover, Deep Q-ICAN preserves NDN semantics by operating without disclosing the identities of consumers or producers. This stands in contrast to several state-of-the-art solutions that perform detection by identifying consumers or specific NDN nodes, thereby compromising the intrinsic privacy guarantees of the NDN architecture and its endpoints.

Furthermore, Deep Q-ICAN effectively neutralizes LDA and FLA attacks by operating at the Interest packet level, making discard or process decisions based solely on the incoming Interests. This approach inherently defeats the attack vectors exploited by LDA and FLA, which rely on manipulating the CS cache content.

As for FLA, Deep Q-ICAN's ability to detect patterns indicative of an FLA within the Interest stream allows it to selectively discard these malicious requests. Because the decision is based purely on the Interest packet, the attacker's strategy of overwhelming the system with a false unpopular unique set of malicious content becomes ineffective at the point where Deep Q-ICAN is deployed.

In essence, by focusing exclusively on the Interest packet flow and making decisions independent of the CS content, Deep Q-ICAN negates the fundamental mechanisms through which LDA and FLA attempt to disrupt NDN network performance and cache efficiency.

However, Deep Q-ICAN may be a bit greedy in terms of memory usage and computational time, depending on the complexity of the scenarios.

In summary, while Deep Q-ICAN effectively addresses key limitations of existing solutions, such as pre-impact detection, privacy preservation, and improved CPA attack detection and mitigation effectiveness, it introduces trade-offs in terms of memory usage during the inference phase. Specifically, the integration of a deep neural network for enhanced feature extraction and decision-making inherently requires a larger memory footprint to store the model parameters compared to the simpler Q-learning approach of Q-ICAN. Furthermore, the computational complexity of processing incoming Interest packets through the neural network leads to a higher CPU utilization during real-time operation (see Table 7).

Table 7
Comparative analysis: Deep Q-ICAN and the State-of-the-Art solutions.

Mechanism	Memory overhead	Computational time	Privacy conservation	Detection Pre/Post attack	Accuracy	Topology	Space storage
Deep Q-ICAN	High	High	No leakage	Pre	High	DFN/AT&T	Low
Q-ICAN [3]	Medium	Low	No leakage	Pre	High	DFN/AT&T	Low
ICAN [24]	Low	Low	No leakage	Pre	High	DFN	Low
SVM [25]	Low	Low	No leakage	Post	Medium	XC/DFN	High
BO-CatBoost [22]	High	High	No leakage	Post	Medium	Tree/DFN	High
FRBM learning [20]	High	High	Leakage	Post	High	AS-3967/DFN	High
RBFNN [19]	Low	High	No leakage	Post	Medium	XC/DFN	High
GBDT [21]	Low	High	Leakage	Post	High	Self Made	Low
Game theory [26]	Low	High	No leakage	Post	Medium	AS-3967	High
HHMCE learning [23]	High	High	No leakage	Post	Medium	Manhattan	High
SEND-DL [27]	High	High	Leakage	Post	High	Self Made	High

Table 8
Comparative Analysis of our mechanism with and without Hop Count.

	Deep Q-ICAN with hop count	Deep Q-ICAN without hop count
CHR (%)	98	97
Accuracy (%)	98.87	98
Stabilization time	680 episodes	668 episodes

8. Discussion

This section provides a comprehensive analysis of the Deep Q-ICAN solution, focusing on its limitations and the performance trade-offs observed during its development and evaluation. We begin by assessing the impact of the hop count feature, a debated metric in NDN architectures. We then discuss broader trade-offs and limitations related to classification strategy, topology sensitivity, and computational resource usage.

8.1. Hop count utility

To evaluate the impact of removing hop count information, we reran an experiment in the DFN topology, modifying the state space to exclude hop count-related features. The results indicate that, without hop count, the accuracy decreased slightly (from 98.87% to 98%) and the CHR decreased from 98% to 97%. However, the stabilization time decreases from 680 episodes (with hop count) to 668 episodes (without hop count).

These results strongly suggest that while hop count was initially considered a potentially useful feature, the core effectiveness of our Deep Q-ICAN approach stems from the other components of our state representation, namely the AVG-CHR and the AVG-IAT. The high CHR and ACC achieved without hop count information underscore the significance of these two features in accurately detecting and mitigating CPA attacks. The relatively quick stabilization further indicates the robustness of the learning process even with a reduced state space. The results of this simulation are illustrated in Fig. 13 and Table 8.

8.2. Limitations and trade-offs

While Deep Q-ICAN delivers strong performance improvements over Q-ICAN, particularly in terms of attack detection and adaptability, these gains come with certain limitations and trade-offs. These span multiple dimensions, including classification priorities, topological scalability, and computational efficiency.

First, we prioritize recall over precision in our model's decision-making. This means Deep Q-ICAN is tuned to favor detecting as many CPA attacks as possible, even at the cost of a slightly higher false positive rate. This trade-off reflects a security-first approach that is especially important in proactive defense systems, where failing to detect an attack is costlier than overreacting to benign traffic.

Second, our evaluation highlights a topology-sensitive performance dynamic. Deep Q-ICAN achieves its strongest results in complex network topologies such as AT&T, where its deep learning capabilities enable it to capture intricate traffic patterns and adapt to diverse operating conditions. In contrast, in simpler or more deterministic topologies like DFN, its performance gains are more modest compared to a tabular Q-learning–based approach like Q-ICAN, and it tends to introduce more false positives due to overestimation bias and aggressive exploration.

Third, there is a trade-off between computational resource usage and detection performance. Integrating a DQN architecture increases memory consumption and training time, due to the larger number of parameters and deeper network structure. However, these costs are justified by substantial gains in performance. For example, Deep Q-ICAN achieves an AVG-CHR of 80% and an attack detection accuracy of 98.87%, compared to Q-ICAN's 72.55% and 96.1%, respectively. This illustrates a favorable resource-performance trade-off, where added complexity enables stronger detection capabilities and more stable convergence.

9. Conclusion and future works

Named Data Networking offers a transformative data-centric architecture with inherent advantages such as efficient content retrieval, stateful forwarding, and in-network caching. However, Cache Pollution Attacks directly threaten these strengths by exploiting caching mechanisms to degrade network performance, increase retrieval latency, and erode trust in content authenticity. If left unmitigated, such attacks risk compromising NDN's scalability and its adoption in latency-sensitive or mission-critical environments, where reliable data availability is paramount. Despite various proposed strategies to counter CPA, many solutions exhibit limited accuracy when facing diverse and unpredictable attack scenarios. Moreover, certain mechanisms rely on consumer identification, contradicting NDN's fundamental principle of identity preservation. Additionally, some methods can overload and exhaust the resources of NDN routers. In this paper, we propose a DRL-based approach for the efficient CPA detection and mitigation in NDN. Deep Q-ICAN leverages Deep Q-learning to comprehend its environment, adapt to evolving patterns, and continuously refine its behavior. Its use of Deep Neural Networks to approximate Q-values enables effective handling of high-dimensional state spaces. Our novel mechanism is an online and efficient mitigation system that outperforms many state-of-the-art mechanisms without requiring a pre-defined dataset for training. We extensively evaluate the performance of our solution, using complex, realistic, and large-scale topologies. Results confirmed that Deep Q-ICAN not only excels in detecting attacks with high accuracy but also demonstrates promising outcomes in increasing CHR and reducing content retrieval delay. In future work, we aim to investigate a wider range of attacks, including Interest Flooding Attack and Cache Privacy Attack.

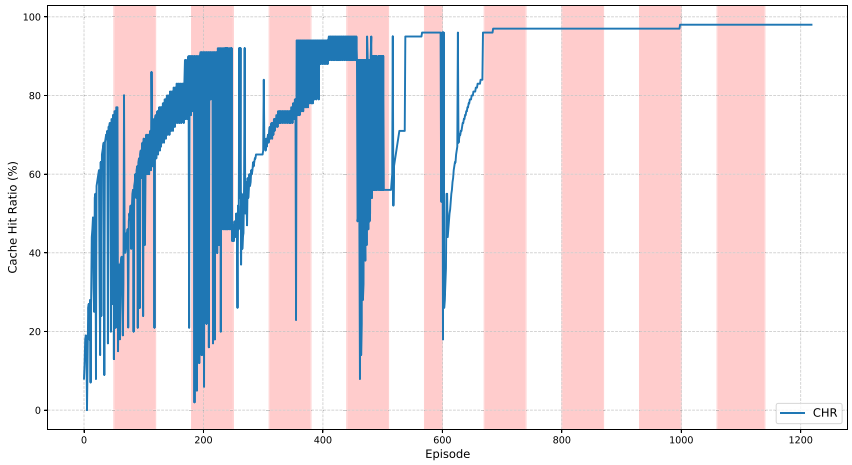


Fig. 13. CHR Evaluation without including the Hop Count in the state space (DFN topology).

Abbreviations

The following abbreviations are used in this manuscript:

DDoS	Distributed Denial-Of-Service
DNS	Domain Name System
PIT	Pending Interest Table
CS	Content Store
FIB	Forwarding Information Base
CPA	Cache Pollution Attack
LRU	Least Recently Used
LFU	Least Frequently Used
FLA	False-Locality Attack
LDA	Locality-Disruption Attack
CHR	Cache Hit Ratio
ARD	Average Retrieval Delay
ICAN	Intrusion detection system for Cache pollution Attack in Ndn
Q-ICAN	Q-Learning based Intrusion detection system for Cache pollution Attack in Ndn
Deep Q-ICAN	Deep Q-Learning based Intrusion detection system for Cache pollution Attack in Ndn
ReBac	Relationship-Based Access Control
RBFNN	Radial Basis Function Neural Network
SVM	Support Vector Machine
GBDT	Gradient Boost Decision Tree
BO-CatBoost	Bayesian Optimization and CatBoost
FRBM	Fuzzy Restricted Boltzmann Machine
HHMCE	Hybrid Heterogeneous Multi-Classifier Ensemble
SEND-DL	Secure Named Data networking using Deep Learning
AVG-CHR	Average Cache Hit Ratio
AVG-IAT	Average Interest Inter-Arrival Time
RTT	Round-Trip Time
ReLU	Rectified Linear Unit
FPR	False Positive Rate
FNR	False Negative Rate
ACC	Accuracy
TPR	Sensitivity
F1	F1 Score

CRediT authorship contribution statement

Abdelhak Hidouri: Writing – review & editing, Writing – original draft, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Haifa Touati:** Writing – review & editing, Validation, Supervision, Methodology, Conceptualization. **Mohamed Hadded:** Writing – review & editing, Validation, Supervision, Conceptualization. **Mohamed Amin Asri:** Software, Conceptualization. **Nasreddine Hajlaoui:** Writing – review & editing, Validation, Supervision, Conceptualization. **Paul Muhlethaler:** Writing – review & editing, Validation, Supervision, Conceptualization. **Samia Bouzefrane:** Writing – review & editing, Validation, Supervision, Conceptualization.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Mohamed Hadded reports financial support was provided by IRT SystemX. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This research work has been carried out jointly at IRT SystemX, IReSCoMath Research Lab and INRIA of Paris, as part of the EXPLO project, which has received funding from IRT SystemX.

Data availability

No data was used for the research described in the article.

References

[1] G. Rajendran, R.S. Ragul Nivash, P.P. Parthy, S. Balamurugan, Modern security threats in the internet of things (IoT): Attacks and countermeasures, in: 2019 International Carnahan Conference on Security Technology, ICCST, 2024, pp. 1–6, http://dx.doi.org/10.1109/ccst.2019.8888399.

[2] H. Touati, A. Aboud, B. Hnich, Named data networking-based communication model for internet of things using energy aware forwarding strategy and smart sleep mode, Concurr. Comput. Pr. Exp. (2021) http://dx.doi.org/10.1002/cpe.6584.

[3] A. Hidouri, H. Touati, M. Hadded, N. Hajlaoui, P. Muhlethaler, S. Bouzefrane, Q-ICAN: A Q-learning based cache pollution attack mitigation approach for named data networking, Comput. Netw. 235 (2023) 109998, http://dx.doi.org/10.1016/j.comnet.2023.109998.

[4] A. Hidouri, H. Touati, M. Hadded, N. Hajlaoui, P. Muhlethaler, S. Bouzefrane, Improving NDN resilience: A novel mitigation mechanism against cache pollution attack, in: 2024 International Wireless Communications and Mobile Computing, IWCMC, Ayia Napa, Cyprus, 2024, pp. 1564–1569, http://dx.doi.org/10.1109/IWCMC61514.2024.10592566.

[5] V. Jacobson, D.K. Smetters, J.D. Thornton, M.F. Plass, N.H. Briggs, R.L. Braynard, Networking named content, in: Proceedings of the 5th International Conference on Emerging Networking Experiments and Technologies, ACM, New York, NY, USA, 2024, http://dx.doi.org/10.1145/1658939.1658941.

[6] H. Asaeda, K. Matsuzono, Y. Hayamizu, H.H. Hlaing, A. Ooka, A survey of information-centric networking: The quest for innovation, IEICE Trans. Commun. (2024) 139–153, https://search.ieice.org/bin/summary.php?id=e107-b_1_139.
[7] A.J. Abu, B. Bensaou, A.M. Abdelmoniem, Dimensioning the pending interest table in content-centric networks, Future Gener. Comput. Syst. 152 (2024) 179–192, http://dx.doi.org/10.1016/j.future.2023.10.009.
[8] Y. Hayamizu, M. Jibiki, M. Yamamoto, Information-centric function chaining for ICN-based in-network computing in the beyond 5G/6G era, IEICE Trans. Commun. (2024) 94–104, https://search.ieice.org/bin/summary.php?id=e107-b_1_94.
[9] N.A. Askar, et al., Forwarding strategies for named data networking based IoT: Requirements, taxonomy, and open research challenges, IEEE Access 11 (2023) 78363–78383, http://dx.doi.org/10.1109/access.2023.3276713.
[10] R. Alubady, M. Salman, A.S. Mohamed, A review of modern caching strategies in named data network: overview, classification, and research directions, Telecommun. Syst. 84 (4) (2023) 581–626, http://dx.doi.org/10.1007/s11235-023-01015-3.
[11] Y. Li, Y. Liu, H. Yin, Z. Guo, Y. Wang, Trident: Defensing synergetic denial-of-service attacks in underwater named data networking, IEEE Internet Things J. 10 (23) (2023) 20633–20648, http://dx.doi.org/10.1109/jiot.2023.3297334.
[12] M.S.M. Shah, Y.-B. Leau, M. Anbar, A.A. Bin-Salem, Security and integrity attacks in named data networking: A survey, IEEE Access 11 (2023) 7984–8004, http://dx.doi.org/10.1109/ACCESS.2023.3238732.
[13] A. Hidouri, M. Hadded, H. Touati, N. Hajlaoui, P. Muhlethaler, Attacks, detection mechanisms and their limits in named data networking (NDN), in: Computational Science and its Applications – ICCSA 2022, 2022, pp. 310–323, http://dx.doi.org/10.1007/978-3-031-10522-7_22.
[14] A. Hidouri, N. Hajlaoui, H. Touati, M. Hadded, P. Muhlethaler, A survey on security attacks and intrusion detection mechanisms in named data networking, Computers 11 (12) (2022) 186, http://dx.doi.org/10.3390/computers11120186.
[15] L. Deng, Y. Gao, Y. Chen, A. Kuzmanovic, Pollution attacks and defenses for internet caching systems, Comput. Netw. 52 (5) (2008) 935–956, http://dx.doi.org/10.1016/j.comnet.2007.11.019.
[16] M. Conti, P. Gasti, M. Teoli, A lightweight mechanism for detection of cache pollution attacks in named data networking, Comput. Netw. 57 (16) (2013) 3178–3191, http://dx.doi.org/10.1016/j.comnet.2013.07.034.
[17] N. Kumar, S. Srivastava, IBPC: An approach for mitigation of cache pollution attack in NDN using interface-based popularity, Arab. J. Sci. Eng. 49 (3) (2023) 3241–3251, http://dx.doi.org/10.1007/s13369-023-07919-1.
[18] H. Yao, M. Li, J. Du, P. Zhang, C. Jiang, Z. Han, Artificial intelligence for information-centric networks, IEEE Commun. Mag. 57 (6) (2019) 47–53, http://dx.doi.org/10.1109/MCOM.2019.1800734.
[19] R.M. Buvanesvari, K. Suresh Joseph, RBFNN: a radial basis function neural network model for detecting and mitigating the cache pollution attacks in named data networking, IET Netw. 9 (5) (2020) 255–261, http://dx.doi.org/10.1049/iet-net.2019.0156.
[20] P.V. Rani, S.M. Shalinie, FuRL: fuzzy RBM learning framework to detect and mitigate network anomalies in information centric network, Sādhanā 45 (2020) http://dx.doi.org/10.1007/s12046-020-01331-3.
[21] D. Man, Y. Mu, J. Guo, W. Yang, J. Lv, W. Wang, Cache pollution detection method based on GBDT in information-centric network, Secur. Commun. Netw. 2021 (2021) 1–10, http://dx.doi.org/10.1155/2021/6658066.
[22] L. Liu, S. Peng, Detection of a novel dual attack in named data networking, in: 2022 IEEE Intl Conf on Parallel; Distributed Processing with Applications, Big Data; Cloud Computing, Sustainable Computing; Communications, Social Computing; Networking, ISPA/BDCloud/SocialCom/SustainCom, IEEE, 2022, pp. 1–8, http://dx.doi.org/10.1109/ispa-bdcloud-socialcom-sustaincom57177.2022.00008.
[23] L. Yao, Z. Zheng, X. Wang, Y. Zeng, G. Wu, Detection of cache pollution attack based on ensemble learning in ICN-based VANET, IEEE Trans. Dependable Secur. Comput. 20 (4) (2023) 3287–3298, http://dx.doi.org/10.1109/tdsc.2022.3196109.
[24] A. Hidouri, H. Touati, M. Hadded, N. Hajlaoui, P. Muhlethaler, A detection mechanism for cache pollution attack in named data network architecture, Adv. Inf. Netw. Appl. (2022) 435–446, http://dx.doi.org/10.1007/978-3-030-99584-3_38.
[25] Y. Cao, D. Wu, M. Hu, S. Chen, Detection and defense schemes for cache pollution attack in content-centric network, Emerg. Netw. Archit. Technol. (2022) 614–629, http://dx.doi.org/10.1007/978-981-19-9697-9_49.
[26] Z. Du, M.S. Obaidat, G. Wu, K.-F. Hsiao, An efficient caching security approach for content-centric mobile networks in internet of things systems, Secur. Priv. (2023) http://dx.doi.org/10.1002/spy2.294.
[27] V.J. Babu, V. Jose, Malicious source detection and threats mitigation in named data networking using deep learning, Int. J. Intell. Eng. Syst. 17 (5) (2024) 440–449, http://dx.doi.org/10.22266/ijies2024.1031.34, 2024.
[28] A. Plaat, Deep Reinforcement Learning, Springer Nature Singapore, Singapore, 2022, p. 2022, http://dx.doi.org/10.1007/978-981-19-0638-1.
[29] A. Hidouri, M. Hadded, N. Hajlaoui, H. Touati, P. Muhlethaler, Cache pollution attacks in the NDN architecture: Impact and analysis, in: 2021 International Conference on Software, Telecommunications and Computer Networks, SoftCOM, 2021, pp. 1–6, http://dx.doi.org/10.23919/softcom52868.2021.9559049.
[30] nsnam, Ns-3, 2024, https://www.nsnam.org. (Accessed 30 December 2024).
[31] L. Campanile, M. Gribaudo, M. Iacono, F. Marulli, M. Mastroianni, Computer network simulation with ns-3: A systematic literature review, Electronics 9 (2) (2020) 272, http://dx.doi.org/10.3390/electronics9020272.
[32] Ndnsim documentation — ndnsim documentation, 2022, https://ndnsim.net/current. (Accessed 30 December 2024).
[33] L. Zahedi, F.G. Mohammadi, S. Rezapour, M.W. Ohland, M.H. Amini, Search Algorithms for Automated Hyper-Parameter Tuning, arXiv.org. https://arxiv.org/abs/2104.14677v1.
[34] M. Tokic, Adaptive ϵ-greedy exploration in reinforcement learning based on value differences, in: KI 2010: Advances in Artificial Intelligence, Springer Berlin Heidelberg, 2010, pp. 203–210.
[35] H. Van Hasselt, A. Guez, D. Silver, Deep reinforcement learning with double Q-learning, Proc. AAAI Conf. Artif. Intell. 30 (1) (2016) http://dx.doi.org/10.1609/aaai.v30i1.10295.
[36] J. Fan, Z. Wang, Y. Xie, Z. Yang, A theoretical analysis of deep Q-learning, Proc. Mach. Learn. Res. 120 (2020) 486–489.
[37] S. Zhang, M. King, X. He, A deeper look at deep Q-learning, 2018, arXiv preprint arXiv:1804.05409.
[38] Q. Lan, Y. Pan, J. Luo, A.R. Mahmood, Memory-efficient reinforcement learning with value-based knowledge consolidation, 2022, In arXiv [cs.LG]. http://arxiv.org/abs/2205.10868.
[39] M.R.R. Tushar, S. Siddique, A memory efficient deep reinforcement learning approach for snake game autonomous agents, 2023, In arXiv [cs.AI]. http://arxiv.org/abs/2301.11977.

Abdelhak Hidouri has completed his Bachelor of Technology (Computer Science) from the Faculty of Sciences of Gabes - Tunisia. He obtained his M.S. degree in computer science and networking from the Faculty of Sciences of Gabes - Tunisia. He is pursuing his Ph.D. degree in Computer Science at the National School for Computer Science (ENSI) - Tunisia and Conservatoire National des Arts et Métiers (CNAM) Paris - France. Abdelhak Hidouri is also a member in the IResCoMath Lab, Gabes - Tunisia and Cedric Laboratory, Paris - France. His areas of expertise are Named Data Networking (NDN), networking and security, information security, and future Internet technologies.

Haifa Touati received her Ph.D. in Computer Science from the National School of Computer Science (ENSI-Tunisia) in 2011, and her HDR in 2021. She obtained her engineering and master's degrees in networking from the same institution. She is currently an Associate Professor of Computer Science at the Faculty of Sciences of Gabes (FSG-Tunisia), where she was the supervisor of the Master's degree program in Computer Science and Networking. Additionally, she serves as the Director of the IReSCoMath research laboratory. Her research interests include named data networking, vehicular communications, security, Machine Learning and blockchain technology. She serves as a TPC member at several international conferences on networking and security.

Mohamed Hadded joins Abu Dhabi University as an assistant professor of cybersecurity engineering, in the College Engineering/CSIT Department. In 2016, he completed his Ph.D. in computer science Engineering at Telecom Sud Paris college in co-accreditation with Sorbonne University (Pierre and Marie Curie Campus). Prior to joining ADU, Dr. Hadded worked as a senior cybersecurity researcher for intelligent transportation systems at IRT SystemX Paris, France from 2021 to 2022. From 2018 to 2022, he also served as an adjunct assistant professor in the Department of Networks and Cybersecurity at Gustave Eiffel University (France). From 2018 to 2021, he worked as a cybersecurity research engineer at VEDECOM institute (Versailles, France). Before that, he worked as a postdoctoral research fellow at the national institute for research in digital science and technology

(INRIA, France) from 2017 to 2018. From 2015 to 2017, he worked as a teaching and research assistant (ATER) at Paris 5 and Franche-Comté (UFC) universities. Since 2021, he serves as a Guest Editor on Wireless Communication Technologies in Intelligent Transport Systems at Electronics Journal and a TPC member at several international conferences and workshops on wireless and mobile networking, Telecommunication, and security.

Mohamed Amin Asri has completed his engineering degree in Software Engineering from the National School of Computer Science and Systems Analysis (ENSIAS) in Morocco, followed by a Master of Science in Artificial Intelligence from Claude Bernard University Lyon 1 in France. He worked on reinforcement learning topics during his final year internship, contributing to research in artificial intelligence. He is currently working as a data engineer, specializing in data processing, integration, and analysis for AI and big data applications. His areas of expertise include software engineering, artificial intelligence, and data engineering.

Nasreddine Hajlaoui received his engineer degree in Networks and Communications from the National School of Engineers of Gabes (Tunisia), in 2007 andresearch M.S. degree in telecommunications from Higher School of Communications of Tunis (Sup'Com) in 2009. He received his Ph.D.degree in Engineering ComputerSystems from the University of Sfax, Tunisia, in2016. He is currently working as assistant professor at Qassim University (KSA) and a member of Hatem Bettaher IResCoMath Lab. His research areas include wireless networking, cloud computing, security and analytical modeling.

Paul Muhlethaler started at Inria (French National Research Institute in Computer Science) in 1988 where he is now a research director. His research topics focus on protocols for networks, with a specialty in wireless networks. He has worked extensively at ETSI and IETF for the HiPERLAN and OLSR standards. He was one of the authors of the first draft of the OLSR protocol in 1997 and co-author of OLSR v1 standard. He was the first researcher to carry out optimizations of CSMA protocols in Multihop Ad Hoc Networks, thereby highlighting the importance of such optimizations. With F. Baccelli and B. Blaszczyszyn, he designed a complete model of an Aloha multihop ad hoc network which led to the design of one of the first multihop ad hoc network offering a throughput scaling according the Gupta and Kumar's famous law. In 2004, he received the prestigious "Science and Defense" award for his work on Mobile Ad Hoc Networks.

His current activity concerns models and performance evaluations especially in wireless and vehicular ad hoc networks. He has also started to study the use of Machine Learning in wireless networks.

Samia Bouzefrane received the Ph.D. degree in computer science from the University of Poitiers, France, in 1998. After four years at the University of Le Havre, France, she joined the CEDRIC Lab of Conservatoire National des Arts et Métiers (Cnam), Paris, in 2002. She is currently Professor in Cnam. She is the coauthor of many books (Operating Systems, Smart Cards, and Identity Management Systems). She has coauthored more than 120 technical articles. Her current research interests include the Internet of Things, vehicular networks, and security using AI techniques. Since 2019, she has been partly delegated to the French Ministry of Higher Education and Research.