Cache Pollution Attacks in the NDN Architecture: Impact and Analysis

Abdelhak Hidouri, Mohamed Hadded, Nasreddine Hajlaoui, Haifa Touati, Paul Mühlethaler

Cache Pollution Attacks in the NDN Architecture: Impact and Analysis

Abdelhak Hidouri
Hatem Bettaher IResCoMath Lab
University of Gabes, Tunisia
abdelhakhdr@gmail.com

Mohamed Hadded
Institut VEDECOM
23 bis Allée des Marronniers,
78000 Versailles, France
mohamed.elhadad@vedecom.fr

Nasreddine Hajlaoui
Hatem Bettahar IResCoMath Lab
University of Gabes, Tunisia
nasreddine.hajlaoui@fsg.rnu.tn

Haifa Touati
Hatem Bettahar IResCoMath Lab
University of Gabes, Tunisia
haifa.touati@cristal.rnu.tn

Paul Muhlethaler
INRIA Paris
2 Rue Simone IFF,
75012 Paris, France
paul.muhlethaler@inria.fr

***Abstract*—Named Data Networking (NDN), one of the most suitable candidates for the future Internet architecture, allows all network nodes to have a local cache that is used to serve incoming content requests. Content caching is an essential component in NDN: content is cached in routers and used for future requests in order to reduce bandwidth consumption and improve data delivery speed. Moreover, NDN introduces new self-certifying contents features that obviously improve data security and make NDN a secured-by-design architecture able to support an efficient and secure content distribution at a global scale. However, basic NDN security mechanisms, such as signatures and encryption, are not sufficient to ensure security in these networks. Indeed, the availability of the Data in several caches in the network allows malicious nodes to perform attacks that are relatively easy to implement and very effective. Such attacks include Cache Pollution Attacks (CPA), Cache Privacy Attacks, Content Poisoning Attacks and Interest Flooding Attacks. In this paper, we identify the different attack models that can disrupt the NDN operation. We conducted several simulations on NDNSim to assess the impact of the Cache Pollution Attack on the performance of a Named Data Network. More precisely, we implemented different attack scenarios and analyzed their impact in terms of cache hit ratio, data retrieval delay and hit damage ratio.**

***Index Terms*—Named Data Networks, Security, Cache pollution attack, NDNSim**

I. Introduction and motivation

Since its proposal in 2010, the Named Data Networking (NDN) architecture has attracted significant attention from the research community that is continuously exploring its potential in various network environments and applications such as WSN [1], IoT [2], [3], VANETs [4], Web [5], [6], Big Data delivery [7] and edge and cloud computing [8]. Moreover, over the last 5 years, several trial deployments have been launched to boost the adoption of NDN by industry, examples being Genomics datasets access based on NDN and the N-DISE project to deploy an NDN petascale data distribution system to serve major science programs [9].

All these applications deal with the exchange of huge and sensitive data, which raises the need for a secure and efficient communication model. However, although NDN is resilient to most traditional security attacks in the TCP/IP model by enforcing the producer to sign each content item before spreading it in the network [10], this architecture remains vulnerable to new types of attacks. With the remarkable increase in security threats and the growing interest in the NDN communication model, securing this architecture against attacks is therefore vital, since fake messages or malicious node behaviour may affect data privacy and disrupt the caching process, which is a core pillar of NDN. Thus, studying the impact of attacks in order to find out how much they can affect NDN operation is essential before forming an appropriate mitigation solution.

This paper outlines the potential threats that could affect the NDN architecture. New vulnerabilities, have been illustrated in this work. Most of these vulnerabilities result from the availability of the data in the cache of intermediate routers and threaten the caching process of the NDN architecture. The most important of these attacks are the Cache Poisoning Attack, the Cache Privacy Attack and the Cache Pollution Attack, in which a malicious node could fake the popularity of the content residing in the cache, deplete the main resources of the legitimate nodes and exhaust them. In a second step of this work, we assess the impact of the Cache Pollution Attack on the efficiency of the caching functionality of NDN in terms of Cache Hit Ratio, Hit Damage Ratio and Data Retrieval Delay. The remainder of this paper is organized as follows: Section II presents the basic concepts and architecture of NDN. In Section III, we describe the security vulnerabilities that can affect the operation of Named Data Networking. Then, in Section IV, we present the simulation results and analyze the impact on NDN performance of the Cache Pollution Attack. Finally, in Section V, we conclude the paper and highlight some open issues.

II. Overview of the Named Data Networking Architecture

In this section, we briefly describe the main building blocks of the NDN communication model.

A Named Data Network (NDN) is a recently proposed Internet architecture that shifts the Internet communication paradigm from the IP address-based model to a content name-based model where consumers request the desired data by names rather than using the IP addresses of the receivers. As explained in RFC 7927 [11], users insert the name of the requested content into a request packet, called an *"Interest"*, and the nearest node that has previously stored this content, or *"Data"* packet, must deliver it. The node that sends the *Interest* is called the *Consumer* and the original *Data* source is called the *Producer*.

To manage the Interest/Data communication process, each NDN node implements three data structures: (i) the *Pending Interest Table (PIT)*, which tracks the outstanding Interests as well as the interface from which they come, in order to deliver the Data back to the Consumer on the reverse path of the Interest, (ii) the *Content Store (CS)*, which stores the received Data packets in order to serve the upcoming Interests that request the same content and (iii) the *Forwarding Information Base (FIB)*, which stores the Data name prefixes and the interfaces to which the *Interest* packets [12] should be forwarded.

To disseminate *Interest* and *Data* packets, each NDN node performs the following process: when an NDN router receives an Interest packet, it searches for the requested Data in its Content Store; if it exists, the Data will be returned through the same interface from which the Interest was received. Otherwise, the router performs a lookup in the PIT. If an entry is found, the router concatenates the interface which received the interest to the PIT entry, otherwise, i.e. if no match is found in the PIT, the router transmits the interest using the information contained in the *FIB*. When an NDN router receives a Data packet, it checks the *PIT* table. If a matching is found, the router forwards the data packet to all interfaces listed in the PIT entry and possibly caches the data in the *Content Store*, otherwise the received Data packet is rejected [13].

III. Security vulnerabilities in Named Data Networks

One key feature of an NDN is that security and privacy are built into the architecture. NDN introduces new mechanisms of signature and cryptography that protect the data from being intercepted by malicious users. Each Data packet is cryptographically signed by its producer in order to build a secure data transmission [14]. With this signature field, consumers and neighbour nodes can verify data integrity and provenance.

Another key feature of NDN is Data caching. As explained in Section II, the Content Store (CS) essentially caches the data received for future requests by the same consumer or even other neighbour nodes, so that the data can be accessed more quickly. Caching in NDN is managed by different policies such as: Least Recently Used (LRU), Least Frequently Used (LFU), First In First Out (FIFO), Random. However, since no pre-existing mechanism is defined to detect the access of different malicious nodes into the Content Store (CS) [15], this "secured-by design" architecture is vulnerable to attacks that mainly aim to abuse the Content Caching process, disturb the availability of the data, maximize the data retrieval delay and manipulate the privacy of data. These attacks include Cache Privacy Attacks, Content Poisoning Attacks, Interest Flooding Attacks and Cache Pollution Attacks. The main principles of these attacks are described in the following subsections.

A. Cache Privacy Attack

In a Cache Privacy Attack, the attacker tries to discover the content items existing in the cache and enumerate them. This step is called an Object Discovery Attack (ODA) [16]. After that, the attacker starts requesting them consecutively. Another type of Cache Privacy Attack is called the Data Flow Cloning Attack (DFCA) [16] where the attackers have knowledge about the content existing in the cache, either by using ODA or by guessing the structure of the name space of the on-going data exchange such as for Voice-over-CCN. Then the attacker tries to clone and predict the next requests.

B. Content Poisoning Attack

In a Content Poisoning Attack, the attacker, presented in most cases as a malicious router, many compromised routers or a malicious producer, has the role of spreading malicious or corrupted data into the neighbour routers which leads to reserving most of the content cache space, and every time a consumer requests such data it spreads through neighbour routers. Although the routers can not check the validity of these malicious data due to their limited resources and time, as a result, only the corrupted content will stay in the cache of the CS, which causes a high delay in retrieving certain content and legitimate content will not be served from the cache. Content Poisoning Attack affects consumer applications and routers [15].

C. Interest Flooding Attack

The fundamental goal of the Interest Flooding Attack is to deplete the resources of the routers. In this type of attack, the attackers send a massive amount of Interests into the targeted node, which prevents legitimate consumers from allocating an entry in the Pending Interest Table (PIT) since the attackers have reserved all the entries in the PIT of the router. This results in Interest timeout and unnecessary retransmissions. There are three types of Interest Flooding Attack suggested by Signorello et al. [17] and Salah et al. [18] in which they use either existing content, or non-existing content (i.e. to confuse the router from detecting the attack), or both types of content.

D. Cache Pollution Attack (CPA)

In a Cache Pollution Attack (CPA), the attackers try to deplete the size of the cache presented in the Content Store

(CS) and deny other legitimate consumers from getting the desired content from the cache. The attackers start requesting unpopular content from a malicious producer that serves these requests. These content items reserve spaces in the content cache for subsequent requests, and they keep requesting the same content, which consumes the total space of the cache and turns these unpopular content items into popular content items that remain in the cache for a longer duration. As a result, they confuse the caching policy and lead to a decrease in the Cache Hit Ratio (CHR) for the targeted routers and increase the data retrieval delay for the legitimate consumers. This type of attack is easy to perform but hard to detect as it varies the content served from the cache and the time to perform the attack is unstable, which, in some cases, exhausts the router and depletes most of the Content Store (CS) resources.

To study the security level of the NDN architecture, in this work we focus on the CPA attack, and we inject it into the NDN architecture to study its impact. The simulation results are detailed in Section IV.

IV. Simulation scenarios and impact analysis of the cache pollution attack

A. CPA simulation setup

To evaluate the impact of the CPA attack described in the previous section, we developed several attack scenarios by injecting malicious behavior into the network. We simulated these attack models using the official NDN simulation module for the Network Simulator-3 (ndnSIM [19]). Two topologies are considered in this study:

i a Simple Network topology in which 3 legitimate consumers, *LConsumer*, request Interests from a legitimate producer, *LProducer*, while an attacker issues Interests to retrieve Data from a malicious producer, *MProducer*, in order to disrupt the cache locality of intermediate routers, $R1$ and $R2$ (see Figure 1).

ii the German Research Network (DFN) [20] which is a German national research and education network (NREN) used for academic and research purposes. The DFN topology allows us to provide a more realistic assessment of the impact of a CPA attack, (see Figure 2).

In all the simulations, the consumers' and attackers' requests follow a Zipf-like distribution. The simulation parameters used in these experiments are summarized in Table I and Table II. To evaluate the performance of the NDNs under the CPA scenarios, we used the following metrics:

- the *Cache Hit Ratio (CHR)* is defined as the ratio of cache hits to the total number of received requests, and it is calculated as follows:

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

- the *Hit Damage Ratio (HDR)* is defined in [21] as the key measure of the effectiveness of a caching attack and is calculated as follows :

$$HDR = 1 - \frac{CHR(without\_attack) - CHR(under\_attack)}{CHR(without\_attack)}$$

When the HDR becomes higher, the performance of the attack increases and when it gets lower the efficiency of the attack decreases, in other words when the HDR is close to 1, the cache is completely under attack and when it is near 0, the attack is completely ineffective.

- the *Average Retrieval Delay (ARD)* is defined as the average delay required for a given LConsumer to receive a requested data packet and is calculated as follows:

$$ARD_k = \frac{\sum_{i=1}^{Nb\_Packets_k} (DRT_k(i) - IST_k(i))}{Nb\_Packets_k}$$

Where $Nb\_packets_k$ is the total number of data packets received by LConsumer k, $IST_k(i)$ is the $Interest\_Sending\_Time$ and it refers to the time when LConsumer k sends the i^{th} Interest packet and $DRT_k(i)$, $Data\_Reception\_Time$, is the time when LConsumer k receives the i^{th} requested data packet.

TABLE I
SIMULATION PARAMETERS FOR THE SIMPLE NETWORK TOPOLOGY

Simulation time	$20s$, $100s$
Number of Legitimate Consumers	3
Number of Attackers	1
Consumer type	ConsumerZipfMandelbrot
Interest rate	40, 80, 120, 160, 200, 240, 280, 320, 360, 400 Interest/s
Routers CS size	50
Cache policy	LRU, LFU, Random, FIFO

TABLE II
SIMULATION PARAMETERS FOR THE DFN TOPOLOGY

Simulation time	$20s$
Number of Legitimate Consumers	6
Number of Attackers	5
Number of Producers	2
Consumer type	ConsumerZipfMandelbrot
Interest rate	120 Interest/s
Attack Time	from $t = 5s$ to $t = 15s$
Routers CS size	50
Cache policy	LRU

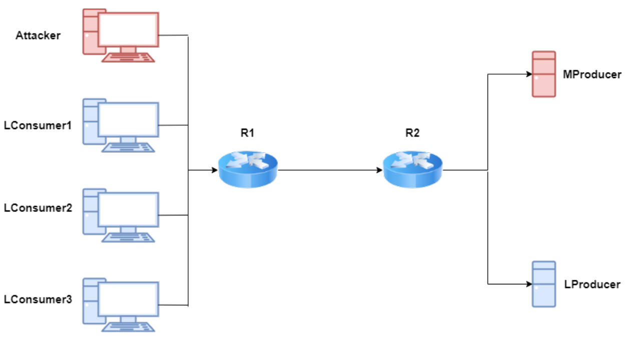


Fig. 1. Simple network topology

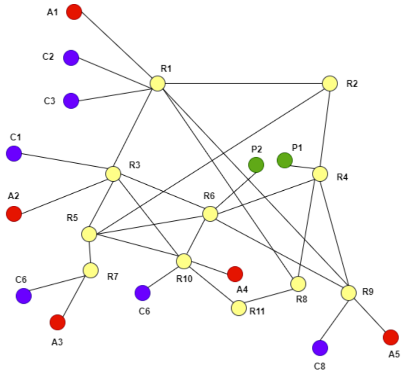


Fig. 2. DFN topology

B. *Impact analysis of the CPA on the Cache Hit Ratio*

To evaluate the impact of the attack in terms of the *CHR*, we use the following three main scenarios:

- *CHR in the simple topology using LRU:*In this scenario, we run simulations for $20s$ and then for longer period, i.e. $100s$, while increasing the Interest sending from 40 to 400 Interest/s. As shown in Figure 3, without attack, every time we raise the frequency of the Interest packets, the CHR rises. For instance, when the LConsumers send 40 interests per second, the CHR reaches 42%, and it keeps increasing, up to 74.41% when the consumers send 400 interest/s.

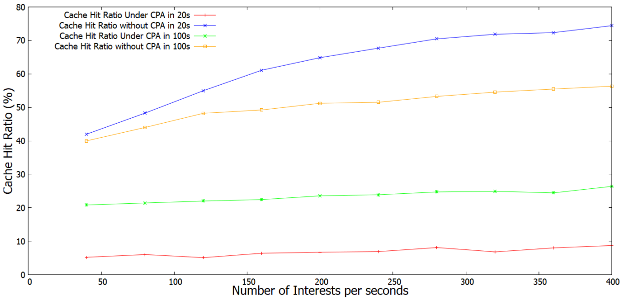


Fig. 3. Cache Hit Ratio using LRU in the simple network topology

Figure 3 also highlights the effectiveness of CPA on the CHR. When the simulation runs for 20s, the CPA, even with only one attacker, drops the CHR from 42% to 5.2% when the interest sending rate is set to 40 interest/s and from 74.41% to 8.7% when it is set to 400 interest/s. The CHR decrease is also confirmed for longer simulation time, i.e. 100s. For example, when the sending rate is set to 400 interest/s, the CHR drops from 56.33% to 26.39%.

- *CHR in the simple topology for different caching policies:* In this scenario we varied the caching policies to discover whether a CPA is still effective. We observe (Figure 4) that no matter which caching policy is adopted, the CHR is still affected by the attack. Using LFU, the value of the CHR drops from 54.96% to 34.84% when sending 40 Interest/s. With FIFO, the CPA leads to a decrease from 53.49% to 25.55% when sending 240 interest/s and by using the Random policy, the CHR decreases from 88.37% to 35.5%.

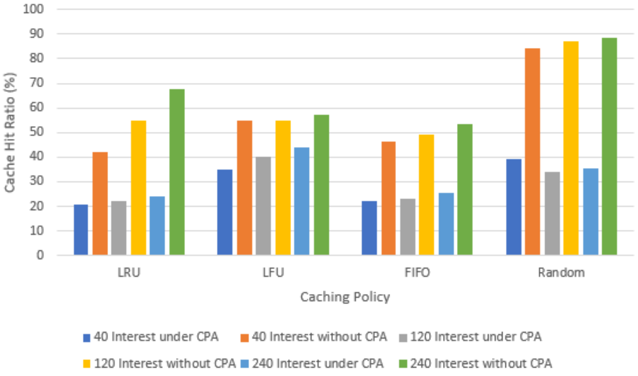


Fig. 4. Cache Hit Ratio using different caching policies in the Simple network topology

- *CHR in the DFN topology:* Figure 5 illustrates, for each edge router, the CHR at each second of the simulation time. Results show that as soon as the CPA starts, i.e. at time $t = 5s$, the CHR takes only one second, i.e. $t = 6s$, to be severely effected by the attack. For example, the CHR of routers $R3$ and $R7$ drops from 41.86% and 28.57%, respectively to about 0%, which means that attackers quickly succeed to completely deprive legitimate consumers from the caching feature benefit.

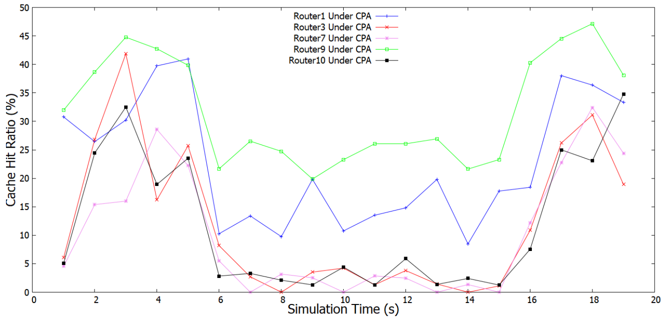


Fig. 5. Cache Hit Ratio of each edge router per time of simulation in the DFN topology

As a result, we can conclude that a CPA has a huge impact on the CHR in both simple topologies and much larger and more complex real-world topologies, which shows the highly negative impact that this attack can have on the availability of the legitimate content in the Content Store.

C. *Impact analysis of CPA on the Hit Damage Ratio*

In this subsection, we analyse the impact of CPA on the HDR, as an essential metric to evaluate the criticality of the attack,

In the simple topology, Figure 6 reports the results obtained for a simulation time set to 20s and 100s. We observe that the HDR reaches the maximum thresh value of 0.365 while in

100s simulation time it gets to the thresh value of 0.5205 by sending only 40 Interest/s.

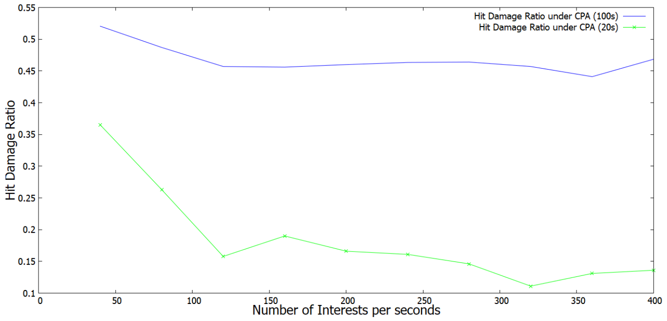


Fig. 6. Hit Damage Ratio in the Simple network topology

Using different cache policies in the simple topology, as shown in Figure 7, we observe that the HDR is still high for each of the caching policies. For example, with LFU, the HDR increases and reaches the thresh of $0,78137$, which demonstrates the huge impact of a CPA on the data served from the Content Cache in a simple topology.

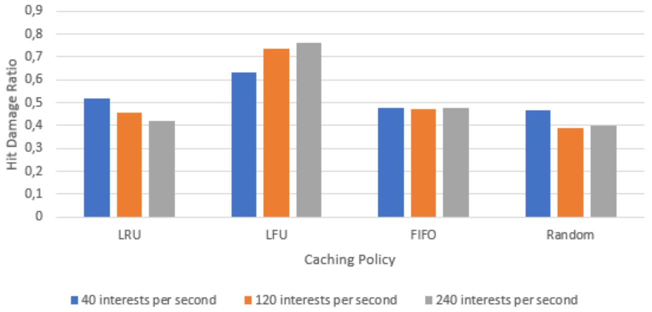


Fig. 7. Hit Damage Ratio for different caching policies in the Simple network topology

In the DFN topology, results plotted in Figure 8 show that by only launching the attack from 5s to 15s the HDR reaches the thresh value of 0.3 in router $R7$ and 0.73 in router $R9$.

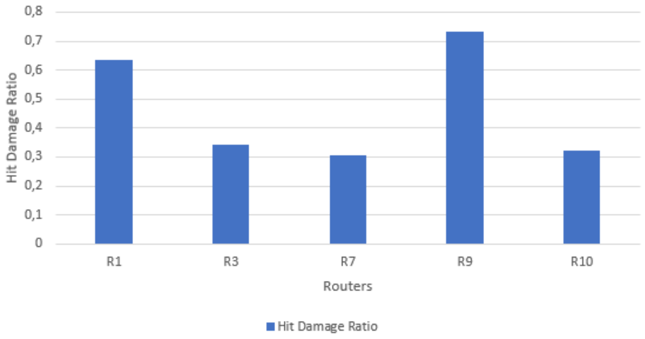


Fig. 8. Hit Damage Ratio in the DFN topology

To summarize, in each of these three main scenarios we observe through the evaluation of the HDR how effective a CPA can be on the content cache, and despite varying the caching policies and the topologies, the attack still has a huge impact on the availability and the popularity of legitimate data in the cache, resulting in diminished performance in Data caching in each of the router nodes.

D. Impact analysis of CPA on Data Retrieval Delay

In this subsection, we evaluate the impact of CPA on the *Average Retrieval Delay (ARD)*, since such an attack also has a negative effect on the legitimate consumers, essentially concerning the delay to retrieve content.

In a simple topology, Figure 9 demonstrates the negative impact of CPA on ARD by measuring it in each of the three legitimate consumers, while varying the simulation time. When the simulation time is set to 20s, the CPA increases the ARD from $14.88ms$ to $16.34ms$ for LConsumer 1, from $14.33ms$ to $15.722ms$ for LConsumer2 and from $14.3ms$ to $15.9ms$ for LConsumer 3. Similar results are obtained for longer simulation time, i.e. 100s. Hence a delay of about $1.5ms$, i.e. an increase of 10% of the average data retrieval delay, is induced by only one cache pollution attacker that is sending 120 interest/s.

In the DFN topology, Figure 10 shows the impact of a CPA on the ARD of 5 legitimate Consumers. We observe that the ARD increases from 15.795ms to 21.19ms for LConsumer 1 and from 15.719ms to 17.84ms for LConsumer 8. In other words, for LConsumer1, the CPA increases the ARD by about $5.48ms$, i.e. 34.88%. Hence we can conclude that the impact of the attack is more severe when the topology gets larger.

To summarize, in both topologies, although the small number of attacker nodes compared to the number of LConsumers, the ARD still becomes higher and higher and this can result in a severe decrease of the perceived throughput of the legitimate consumers, a decrease in the performances of the NDN routers and wastage of the network bandwidth usage.

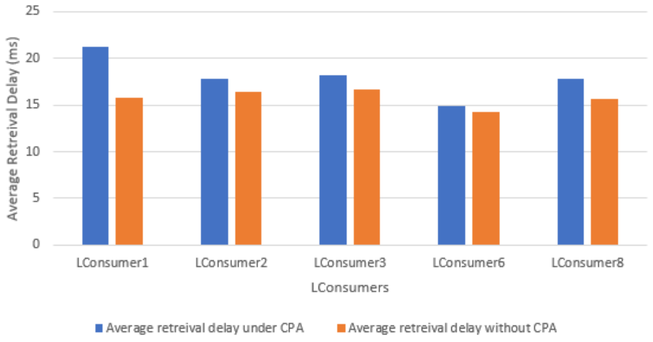
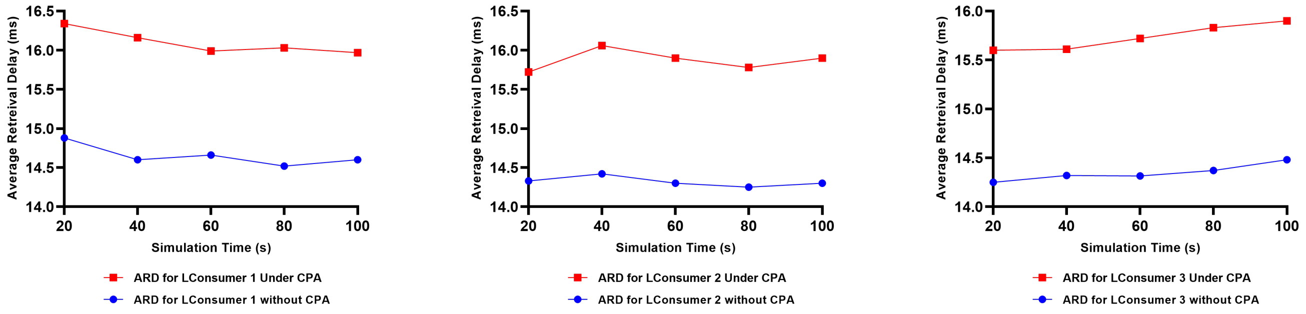


Fig. 10. Average retrieval delay in the DFN topology

V. Conclusion and Future Works

The success of NDN has always been based on its security and its high performances, ensuring it a promising future in the network revolution. However, since NDN is not vulnerable to a range of basic attacks that are effective on the TCP/IP model, a



(a) LConsumer 1

(b) LConsumer 2

(c) LConsumer 3

Fig. 9. Average retrieval delay of each legitimate consumer in the simple network topology

number of new attacks have appeared that target NDN's main components.These attacks include the Cache Pollution attack (CPA), the Interest Flooding Attack, the Content Poisoning Attack and the Cache Privacy Attack. In this paper, we have analysed one of the most effective attack of these attacks, the CPA, and we have studied its impact on NDN through ndnSim simulations. Using different scenarios in simple as well as complex and realistic topologies, we have shown the impact of a CPA on the caching efficiency More specifically, our results reveal that a CPA decreases the CHR to almost 0% in several scenarios and increases the ARD to around 20% while the HDR reaches 0.6, which confirms the highly negative impact of this form of attack. In future work, we intend to exploit the results of this investigation to design a solution for detecting and mitigating the Cache Pollution Attack. In particular, we will develop an intelligent mechanism that computes the illegibility of each cached data packet and uses this parameter to improve the caching policy.

References

[1] Aboud, A., Touati, H.: Geographic Interest Forwarding in NDN-Based Wireless Sensor Networks. In: 2016 IEEE/ACS 13th International Conference on Computer Systems and Applications (AICCSA). pp. 1-8(2016). https://doi.org/10.1109/AICCSA.2016.7945683

[2] Aboud, A., Touati, H., Hnich, B.: Efficient forwarding strategy in a NDN-based internet of things. Cluster Comput. 22(3): 805-818(2019). https://doi.org/10.1007/s10586-018-2859-7

[3] Wang, X., Cai, S.: Secure healthcare monitoring framework integrating NDN-based IoT with edge cloud. Future Gener. Comput. Syst. 112: 320-329 (2020). https://doi.org/10.1016/j.future.2020.05.042

[4] Kardi, A., Touati, H.: NDVN : Named Data For Vehicular Networking. International Journal of Engineering Research Technology (IJERT), 4(4), (2015).

[5] Nan, G., Qiao, X., Tu Y., Tan W., Guo L., Chen J.: Design and Implementation: the Native Web Browser and Server for Content-Centric Networking. Computer Communication Review 45(5): 609-610(2015). https://doi.org/10.1145/2829988.2790024

[6] Qiaoa, X., Rena, P., Chen, J., Tan, W., Blake, M. B., Xu, W.: Session persistence for dynamic web applications in Named Data Networking. Journal of Network and Computer Applications, 125:pp.220-235(2019). https://doi.org/10.1016/j.jnca.2018.10.015

[7] Mejri, S., Touati, H., Kamoun F.: Are NDN Congestion Control Solutions Compatible with Big Data Traffic? In: 2018 International Conference on High Performance Computing & Simulation, (HPCS), pp. 978-984(2018). https://doi.org/10.1109/HPCS.2018.00154

[8] Ullah, R., Rehman, M.A.U., Kim, B.S.: Design and Implementation of an Open Source Framework and Prototype For Named Data Networking-Based Edge Cloud Computing System. IEEE Access, 7:57741-57759(2019). https://doi.org/10.1109/ACCESS.2019.2914067

[9] Touati, H., Mejri, S., Malouch, N. and Kamoun, F., "Fair hop-by-hop interest rate control to mitigate congestion in named data networks". Cluster Comput (2021). https://doi.org/10.1007/s10586-021-03258-8

[10] Mejri S., Touati H. and Kamoun F., Preventing unnecessary interests retransmission in named data networking, International Symposium on Networks, Computers and Communications (ISNCC), 2016, pages 1-6, https://doi.org/10.1109/ISNCC.2016.7746058.

[11] RFC7927, Information-Centric Networking (ICN) Research Challenges, https://datatracker.ietf.org/doc/html/rfc7927, Jul. 2016.

[12] S. Mejri, H. Touati, N. Malouch and F. Kamoun, "Hop-by-Hop Congestion Control for Named Data Networks," 2017 IEEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA), 2017, pp. 114-119, doi: 10.1109/AICCSA.2017.36.

[13] Mejri S., Touati H. and Kamoun F., Hop-by-hop interest rate notification and adjustment in named data networks, IEEE Wireless Communications and Networking Conference (WCNC), 2018, https://doi.org/10.1109/WCNC.2018.8377374

[14] L. Zhang et al., Named data networking, ACM SIGCOMM Comput. Commun. Rev., vol. 44, no.3, pp.66–73, 2014, https://doi.org/10.1145/2656877.2656887

[15] Kumar N., Ashutosh K., Abdul A., Shashank S., Security Attacks in Named Data Networking: A Review and Research Directions. Journal of Computer Science and Technology 34(6):1319-1350, November 2019. http://dx.doi.org/10.1007/s11390-019-1978-9

[16] Lauinger T, Laoutaris N, Rodriguez P, Strufe T, Biersack E, Kirda E., "Privacy Implications of Ubiquitous Caching in Named Data Networking Architectures", Technical Report, Northeastern University, 2012, June 2019. https://tobias.lauinger.name/papers/ccn-cache-attacks-tr-iseclab-0812-001.pdf

[17] Signorello S, Marchal S, Fran¸cois J et al., Advanced interest flooding attacks in Named-Data Networking. In Proc. the 16th IEEE International Symposium on Network Computing and Applications, October 2017, pp.1-10 https://doi.org/10.1109/NCA.2017.8171325

[18] Salah H, Strufe T., Evaluating and mitigating a collusive version of the interest flooding attack in NDN. In Proc. the 2016 IEEE Symposium on Computers and Communication, June 2016, pp.938-945. http://dx.doi.org/10.1109/ISCC.2016.7543857

[19] S. Mastorakis, A. Afanasyev, and L. Zhang, "On the Evolution of ndnSIM: an Open-Source Simulator for NDN Experimentation", ACM Computer Communication Review, July, 2017.

[20] Deutsches forschungsnetz (DFN), https://www.dfn.de/en/

[21] Karami, A., Guerrero-Zapata, "An ANFIS-based cache replacement method for mitigating cache pollution attacks in Named Data Networking", Computer Networks (COMPUT NETW), February, 2015.