Attacks, Detection Mechanisms and Their Limits in Named Data Networking (NDN)

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

Attacks, Detection Mechanisms and Their Limits in Named Data Networking (NDN)

Abdelhak Hidouri[1], Mohamed Hadded[2], Haifa Touati[1], Nasreddine Hajlaoui[1], and Paul Muhlethaler[3]

[1] Hatem Bettaher IResCoMath Lab, University of Gabes, Gabes, Tunisia
abdelhakhdr@gmail.com, haifa.touati@cristal.rnu.tn, hajlaoui.ing@gmail.com
[2] IRT SystemX, France
mohamed.elhadad@irt-systemx.fr
[3] INRIA, Paris, France
paul.muhlethaler@inria.fr

Abstract. Proposals for Information Centric Networking (ICN) have recently emerged to rethink the foundations of the Internet and design a native data-oriented network architecture. Among the current ICN projects, Named Data Networking (NDN) is a promising architecture supported by the National Science Foundation (NSF). The NDN communication model is based on the Publish/Subscribe paradigm and focuses on broadcasting and finding content and introduces caching in intermediate routers. Data packets are sent in response to a prior request called an Interest packet and the data are cached along the way to the original requester. Content caching is an essential component of NDN in order to reduce bandwidth consumption and improve data delivery speed, however, this feature allows malicious nodes to perform attacks that are relatively simple to implement but very effective. For that reason, the goal of this paper is to study and classify the types of attacks that can target the NDN architecture such as (Cache Pollution Attack (CPA), Cache Poisoning Attack, Cache Privacy Attack, Interest Flooding Attack (IFA), etc) according to their consequences in terms of reducing the performance of the network. Moreover, we give an overview about the proposed detection mechanisms and their limitations.

Keywords: Named Data Networking · NDN attacks · Attack Detection.

1 Introduction and motivation

The Internet architecture was developed in the late 70s, when its main goal was only to ensure communication between a limited number of devices. Since the release of the World Wide Web, the original design of the Internet has had to cope with many new requirements such as mobility, security, scarcity of IP addresses, etc. Such requirements pose serious challenges to the traditional TCP/IP architecture. These challenges include the inefficiency of the security model deployed in the TCP/IP architecture to deal with the increasing number of vulnerabilities and attacks that are recorded every day, such as Denial of Service attacks (DoS),

Distributed Denial of Service attacks (DDoS), relay attacks, traffic analysis attacks, flooding attacks, etc. Moreover, massive content distribution has changed data communication in recent years. Today's Internet is mostly characterized by the consumption of multimedia material.

Many solutions have been suggested to deal with the vast amount of data traffic, such as Content Delivery Networks (CDN), Peer to Peer (P2P) networks and Distributed Database (DDB) systems [1]. Most of these solutions are based on the deployment of dedicated servers situated in areas close to consumers to hold content replicas and optimize the download experience for consumers. Nonetheless, the widespread adoption of these solutions is far from being an ideal solution for the content delivery issue. These solutions require ISP cooperation and DNS configurations. Furthermore, they impose significant operational and capital costs that can only be afforded by a small number of large commercial companies. Importantly, these solutions are still vulnerable to multiple security issues.

Recently, a new research direction in networking, called Information-Centric Networks (ICN), has been proposed to deal with massive content distribution. The underlying philosophy of ICN is to focus on the content itself and not on its location. Several ICN architectures have been proposed in the literature [2], mainly in the United States and Europe, and include TRIAD, DONA, PSIRP and its successor PURSUIT, CCN and NDN. Among all these projects, the NDN architecture seems to be the most promising.

NDN is an architecture initially proposed by Van Jacobson in 2009 [3] and supported by the National Science Foundation (NSF). It follows a receiver-based communication model and introduces caching in intermediate routers. Data packets are sent in response to a prior request called an Interest packet and the data are cached along the way to the original requester[4]. In other words, the NDN communication model is based on the Publish/Subscribe paradigm and focuses on content distribution and discovery. This receiver-based service model naturally adapts to the data-centric model of several emerging networks, like WSN[5], IoT[7] and VANET[6].

Content caching is an essential component of NDN and serves to reduce bandwidth consumption and improve data delivery speed. Additionally, NDN introduces new content self-certification (signing) features that obviously improve data security and make NDN a security-by-design architecture capable of supporting efficient and effective content distribution, and large-scale security. However, basic NDN security mechanisms, such as signatures and encryption, are not sufficient to ensure the complete security of these networks. Indeed, the availability of data in several network caches allows malicious nodes to carry out attacks that are relatively simple to implement but very effective.

Unlike the traditional Internet where data is centralized on servers and accessible to the public via TCP/IP, in NDN, routers can cache data and reuse it, reducing the number of repeated requests and the resulting overhead for today's Internet services. However, new attacks specific to the NDN architecture have recently been identified in the literature. In particular, these attacks target the

new internal structures of routers, such as the table used for pooling requests or the cache. These attacks have not yet been precisely characterized, thus leaving the first NDN networks deployed open to a potential danger.

In this paper, we present the relevant attack models that can threaten the NDN architecture and the impact such attacks may have. Then, we give an overview of recent attack-detection mechanisms, and we point out some of their limitations. The remaining part of this paper is organized as follows. In Section 2, we highlight the NDN architecture. In Section 3, we give an overview of security attacks on NDN. Section 4 introduces the state of the art regarding the detection mechanisms and in Section 5, we discuss their limitations. Finally, the conclusion is given in Section 6.

2 Overview of the NDN architecture

In this section we present the main building blocks of the NDN architecture, especially the data structures implemented in each node to manage data dissemination as well as the NDN naming scheme. In NDN, all communications are performed using two distinct types of packets: *Interest* packets and *Data* packets. The node that sends an Interest is called the *Consumer* and the original Data source is called the *Producer*. The Data and Interest packets carry a *name* field, which uniquely identifies a piece of content that can be carried in one Data packet [8]. Each node in NDN implements three components:

- **The Pending Interest Table (PIT):** It contains the interest packet entries and the corresponding list of incoming interfaces and "on hold" Interests that have not yet been satisfied. Multiple incoming interfaces indicate the same Data is requested from multiple downstream users.
- **Forwarding Information Base (FIB)** : FIB entries associate prefix names to one or many output interfaces to which the Interest packets should be forwarded.
- **Content Store (CS):** The CS holds a copy of the *Data* packets that have been passed by the NDN router, and this copy will be used to satisfy subsequent interests that request the same content. This caching mechanism speeds up data delivery and decreases the network load.

3 Security threats in the NDN Architecture

As explained in the previous section, most of the attacks in the current TCP/IP model, such as snooping, spoofing, traffic analysis attack, Man In The Middle, SMB attack, relay attack, etc., are no longer effective in the NDN architecture due to its security foundation. This security foundation is based on signing each data packet by the producer.

An attack called "DNSspooq" [9] had a huge impact on TCP/IP architecture, where it leaked plenty of legitimate users' private information. This attack uses the IP addresses of each user to attack the neighbour users. This attack is no

longer applicable in NDN, because NDN uses naming instead of IPs and encapsulates a signature field. Moreover, this attack could not be applied because NDN relies only on the interfaces to transmit the Interest and the Data packets.

Despite this solid security base in NDN, this secured by design" architecture still suffers from several vulnerabilities, mainly: Cache Pollution Attack (CPA), Cache Privacy Attack, Cache Poisoning Attack and Interest Flooding attack (see Figure 1). These attacks affect its essential components, namely the PIT and the CS, which is very critical since content can be spread everywhere through the pervasive caching of NDN.

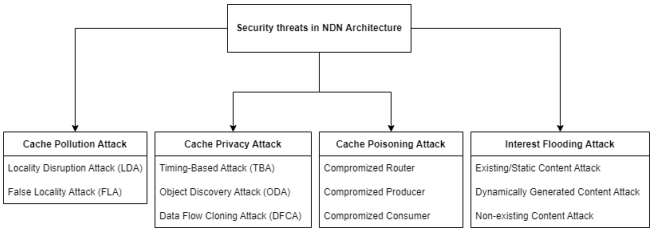

Fig. 1. Security threats in NDN Architecture.

3.1 Cache Pollution Attack (CPA) and its impact

In a Cache Pollution Attack (CPA), the attacker tries to cache unpopular content in the Content Store (CS) in an attempt to make the cache unavailable to legitimate consumers. This attack mostly targets NDN routers Cache Hits.

As shown in (Figure 2), an *Attacker* node sends interest packets to change the priority of the content stored in the CS of nearby routers. This induces the caching of a large number of malicious packets in the CS of router R1 (malicious contents are represented in red in our example). This behaviour changes the priority of the content and increases the popularity of these malicious content items. As a result, this attack reduces the probability of obtaining legitimate content from the cache by legitimate consumers.

This attack is difficult to detect because it is hard to identify the attacker in the NDN architecture (NDN conserves the privacy of the consumers). In addition, the attacker does not follow a specific pattern, such as the amount of interests sent per second and the hierarchy of prefix naming. Moreover, the time of launching the attack is not stable and, in several cases, follows different strike-timing.

Deng et al. [10] classified CPA into 2 main categories : *Locality Disruption Attack (LDA)* and *False Locality Attack (FLA)*. For LDA, the malicious consumer requests masses of junk content. This type of content gets cached in the CS and this malicious consumer keeps requesting it so as to keep it in the CS.

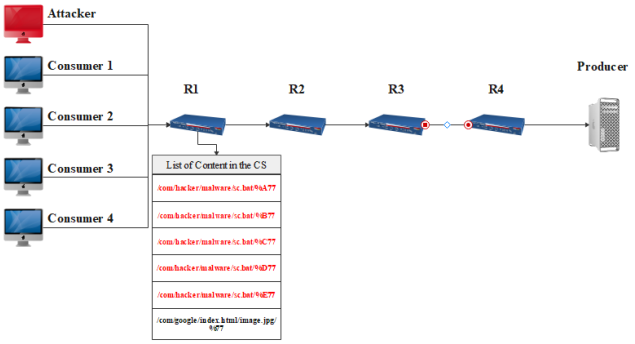

Fig. 2. Cache Pollution Attack (CPA).

In the case of FLA, the malicious node requests already existing content with less popularity in order to prevent other legitimate content from getting higher popularity and falsify the priority rules implemented in the caching policies, which forces it to be useless. As explained in [11], CPA essentially affects the Cache Hit Ratio (CHR) by decreasing the hit ratio of legitimate content. [12] Explains that in several cases the CHR of legitimate requests could decrease to 0% in edge routers, i.e. routers that are directly connected to the attackers. CPA attacks also affect the Average Retrieval Delay (ARD), which greatly increases compared to the normal state. This leads to a direct effect on the legitimate consumer and may result in an unnecessary retransmission of the legitimate interest.

3.2 Cache Privacy Attack and its impact

In a Cache Privacy Attack, the attacker tries to access the cached content present in the CS to find out whether this content has recently been accessed by certain legitimate consumers or not. When the attacker knows the time of the access of such content, the attacker associates this content either to malicious consumers or to different legitimate consumers. This attack breaks the privacy of the legitimate users, and it requires several steps to apply it starting from the "Enumeration" of the content present in the CS and other information like the access time of each content item in the CS cache, then requesting such content by either predicting the next pattern of the interest or requesting the same content.

The authors of [13] studied cache privacy attack for CCN. They identified three types of cache privacy attack, such as a request monitoring attack or Timing-Based Attack (TBA), Object Discovery Attack (ODA) and Data Flow Cloning Attack (DFCA).

In TBA the attacker tries to enumerate the content presented in the CS. The attacker finds the hit time of the cache by requesting the same content twice. The first request caches the content. The second request is satisfied from the cache, as the content has been already cached. The attacker requests the desired

content to check whether it remains in the cache or not. If it is cached and the cache hit occurs by legitimate users, the attacker can interpret that the consumer who requested this content is linked to such content.

In ODA the attacker sends an interest that has the root name space "/", then the router responds with a random content from the cache of the nearby router or any other routers by specifying the HopLimit. After that, it constructs the prefix based on the recent result of the enumeration. Taking the example of an attacker who sends an interest with root name space "/", the CS cache responds with "/com/website/www/media/video.mp4/%77", then the attacker changes the prefix into "/com/website/www/media/" to get a list of content in that name space and so on. In DFCA, the attacker tries to use ODA to enumerate the cache content. He targets the ongoing flow interests and predicts the next name space that the legitimate consumer still hasn't requested and sends it to change the ongoing flow interest on his side. This attack is applied, for example, in Voice-over-CCN applications. Globally, a Cache Privacy Attack targets the confidentiality of content in the CS.

3.3 Cache Poisoning Attack and its impact

A Cache Poisoning Attack mainly aims to inject either fake or corrupted content into the router, which remains in the routers and keeps spreading into the neighbour nodes. Processing a content item in a line speed timing, leads the NDN router to be unable to deeply verify the malicious content [14]. This attack can be performed either by a compromised router which spreads poisoned content in reply to the interest packets. The other neighbour routers will cache this content which will be accessed later by other consumers. The attack is highly dangerous as it can distribute poisoned content through compromised publishers and routers, and spread fake or corrupted content very quickly. The main security concern in a Cache Poisoning Attack is the availability of the content.

3.4 Interest Flooding Attack and its impact

An Interest Flooding Attack is a type of attack that targets the Pending Interest Table (PIT), and aims to send a huge amount of interests into the desired router, which forces all these routers to create entries in their PIT that remain open during the attack. As a result, the PIT entries will no longer be available for legitimate consumers and keep dropping each packet sent by them [11] [14].

Three types of such attacks have been specified : existing or static, dynamically-generated and non-existing attack. For the first type, the attacker sends an interest of an existing content item that will be cached in the CS and this will open a small number of entries in the PIT. The second type is based on dynamically generating different interests with high frequency. This type of attack is more efficient than type 1. For the type 3, the malicious node sends interest packets of non-existing content. This type ensures that the router creates a higher number of PIT entries that remain open until the time-out. This type is more severe

than types 1 and 2 because PIT entries remain open for a longer duration. Table 1 summarizes the four attacks introduced in this section, we compare their security goals, the NDN component affected and the potential attacker entity.

Table 1. The effects of NDN attacks on the security goals

Attack	The attacked entity	Target Security Goal	Target Entity
Cache Pollution	Consumer/Producer	Availability	CS
Cache Privacy	Consumer	Confidentiality	CS
Cache Poisoning	Consumer/Producer	Availability	CS
Interest Flooding	Consumer	Availability	PIT

4 Attack detection mechanisms in NDN

4.1 CPA Detection Mechanisms

The first early mechanism to detect CPA has been proposed by Mengjun Xie et al. in [15] and is called, CacheShield. The main process of this mechanism is based on a shield function which calculates the frequency of receiving a content item and indexing it with each associated content based on its popularity and its naming-prefix, then it verifies if such content is already cashed. If so, it delivers it to the consumer that requests it, otherwise, it stores a slice of this content and resets its frequency counter.

By respecting the chronology, Guo et al. [16] suggested a mechanism based on path diversity. The goal of this mechanism is to base its decision on the collected information related to each data packet stored in the cache of the CS. The collected parameters concisted by the hit time of a content item $o.hit$, the path traversed of a content item $o.PathTracker$, the damage ratios on backbone router, false positive of the decision and false negative error ratios of the decision.

Then this mechanism compares the normal state before the attack and the attack state using a “PathTracker”. The authors defined a threshold which is compared with the value obtained. If the value is higher than the predefined threshold, the attack is detected. Kamimoto et al. [17] suggest a different mechanism based on prefix hierarchy and called "Cache Protection Method based on prefix Hierarchy" (CPMH). This mechanism goes through 3 main steps : extracting the malicious nodes’ prefixes and saving them on a BlackList. This is done by calculating the Request rate Variation for each prefix (RVP), and to avoid obtaining a false positive, the mechanism also calculates the Weighted Request rate Variation for each prefix (WRVP). The next step is to eliminate the prefixes existing on the BlackList. Finally, each interest that requests a blacklisted prefix will be dropped. Zhang et al. [18] propose a mechanism based on the Coefficient of Variation (CV), where the decision is made based on several parameters, namely : The prefix received x, the interface from which the interest is received and the frequency of sending such an interest.

These collected statistics are used to calculate the $CV_i(x)$ of an interest i with prefix x, if the value is low, it is more likely that it is from an attacker source, if not, it is suitable to be cached. The decision in this mechanism is made on the data, which means either to cache it or not to cache it. Another CPA detection mechanism has been proposed in [19] and it is based on clustering. Globally, this mechanism collects some information upon the running of the mechanism, for-instance, the number of interests received, the number of interests for the same content and the time interval between two consecutive requests for the same content

Then it classifies the interests into 2 different clusters based on the probability of an interest i appearing and the AVG time interval between two interests requesting the same content. This helps in the final step to determine the type of the attack (LDA or FLA) then each router broadcasts to its neighbours a list of suspicious content items in order to avoid them being cached.

The authors of [20] suggested a detection mechanism called ICAN (Intrusion detection system for CPA attack in NDN architecture) based on metrics of performance including the average cache hit ratio, average interest inter-arrival time, hop count and prefix variation that is basically stands by monitoring dynamically the variation of those metrics to decide the appearance of the attack in different realistic network topologies. This solution demonstrated high efficiency compared to previous mentioned solutions in terms of conserving router resources, the conservation of the user's privacy and the high accuracy of detecting the attack.

Other detection approaches that are Based on Probability Decision (BPD). This approach includes a mechanism called Randomness Check [21]. This solution starts by creating a matrix composed of the names and how many times each content item is received. Upon receiving some content, it increments the value of that content in the matrix. If requested content goes higher than the predefined threshold, it is suspected of being an attack. In the case of an attack, the content is eliminated, otherwise, it stays for future demand.

A detection mechanism called Adaptive Neuro-Fuzzy Inference System (ANFIS) has been proposed by Karami et al. [22], which mainly aims to change the caching replacement policy. The main step in this mechanism starts by collecting data related to each interest which lead to feeding the features of its neurones that are constituted by 5 fuzzy layers, as result a goodness value is collected on each interest. This value is taken into consideration to decide whether to cache the content or remove it from the cache of the CS.

Kumar et al. [23] proposed a mechanism to detect CPA called Interface-Based Popularity Caching (IBPC) which goes through collecting data based on the number of interfaces that receive a content item in a certain period of time. IBPC focuses on calculating the number of users requesting a content item using the Exponentially Weighted Moving Average (EWMA) to define the popularity of the content over a given period of time, considering that the number of attackers is smaller than the number of legitimate consumers. New technique of detecting CPA has been introduced by Lei et al. [24], where this mechanism

is relies on collecting information such as, Data validator, Provider CS, Content Name, Task Prefix and Digest Prefix.

By collecting this information, the mechanism constructs blocks on each content item and verifies by comparing it with a predefined threshold to either proceed in caching the content or denying it from being accessed to the cache of the CS.

4.2 Cache Privacy Attack Detection Mechanisms

For a Cache Privacy Attack, [25] proposed a mechanism relying on disabling the *scope* and the *exclude* fields because the malicious node is able to perform the attack by utilizing those 2 fields. More precisely, the *exclude* field guarantees that the attacker gets the desired content without getting the same content covered from the first attempt, and the *scope* field allows the attacker to request the root namespace, which makes it easy for him to locate the desired content.

Ntuli et al. [26] suggested checking the interest and the content associated to the same prefix in order to identify the attack probability. The attack is suspected by the increase in the cache hit and the frequency of sending an interest, so it denies such an interest from accessing the content. [27] extended the work presented in [26] by adding other parameters such as repeat requests for multiple content in short period of time. A predefined threshold is used: if the goodness value decreases, an attack is suspected. Kumar et al. [28] proposed to decide either to add a delay on the malicious interest or to let it pass by judging through a static defined prefix hierarchy. This mechanism is applied for a Timing-Based Attack (TBA).

4.3 Cache Poisoning Attack Detection Mechanisms

The detection of Cache Poisoning Attack is hard and expensive in terms of the resources of the NDN router, which led Gasti et al. [29] to propose a mechanism based on Self-certifying the data packet and the interest packet. Certain information is collected, including the hash of the content associated by the mechanism by each passing data, the Content Name and the signature of the content.

This makes it possible to compare both the data collected of the interest and the data packet. Kim et al. [30] extended the work by reducing the overload on the NDN routers caused by these extensive verifications. The mechanism mainly aims to verify only the content existing in the cache: in the case of a cache hit, the verification is applied, otherwise, it is not. Ghali et al. [31] investigated the attack by associating each content item in the cache with a rank, this rank is variated by the number of requests for content x. The lower the value is, the higher probability there is of an attack.

4.4 IFA Detection Mechanisms

In IFA, some mitigation mechanisms have been suggested. In [32], the authors propose using PIT entries as the main parameter when it goes beyond a predefined threshold. Then, the NDN router lists the unwanted interest packets with

the longest prefix and sends the associated data packets to the backbone routers which process the interface where this interest comes from. Compagno et al. [33] proposed a mechanism to detect IFA by utilizing the ratio of interests that come from an interface i, the ratio of interests that go out from an interface $i*$ and PIT capacity for each interface i.

The mechanism calculates the goodness value using those parameters, if it goes more than a pre-defined threshold, an attack is detected so the NDN router revokes such an interest and a notification message is sent to the neighbour router about the malicious interest. A neural network mechanism has been suggested by [34], the features used in detection are : The number of arriving data packets, the number of arriving interest packets, the number of outgoing data packets, the number of outgoing interest packets, the number of satisfied interest packets and the size of the PIT entries.

This mechanism has been evaluated on the DFN topology and has shown high accuracy compared to recent, previously mentioned mitigation mechanisms.

5 Detection mechanisms limitations

As shown in Table 2, most of the detection mechanisms presented in the previous section (II) aim to provide a better strategy to prevent the effect of the attacks presented in Section (III) or to limit their impact. These mechanisms present a variety of vulnerabilities that can be exploited by the attackers to exhaust the router resources and expose the end-point node's identity, etc.

An early solution, CacheShield [15] presented several limitations that affect the efficiency of the NDN routers, such as the complexity of the algorithm for detecting the malicious behaviour of CPA. Another limitation is space exhaustion as this mechanism does not consider the limited space that the cache of the CS has, the name placeholder can cause an issue in this point, in which this mechanism keeps storing them continuously. Prefix Hierarchy suggested by [17] solves the problem of exhausting NDN router resources, but still depends on the topology itself. Also, this mechanism has issues when it comes to performing an attack consisting of caching unpopular content of popular prefix in order to confuse the mechanism. The mechanism proposed by Guo et al. [16], has a medium level of memory exhaustion. In addition, this mechanism is not applicable on all types of topology, and it keeps storing the information collected on each interest, which over-consumes the router space resources. The clustering method recommended by Yao et al. [19] has shown good results in classifying different attack models such as LDA and FLA, but it still has a lack of certainty and makes too many false judgements, where in this case the attacker keeps demanding low popularity content to spoil the cache of the CS. ANFIS [22] has been a more reliable suggestion compared to previous mechanisms in terms of accuracy and efficiency. But, this mechanism may fall into two main states :

– The huge exhaustion of the routers resources such as bandwidth, caching resources, etc. because many data packet need to be cached in order to decide the probability of attack.

– The huge rate of false positives, which leads the mechanism to allow content demanded by the malicious node to be cached hindering the storage demanded by legitimate nodes.

On the other hand, Cache privacy Attacks present a challenge for researchers to invent new mechanisms that can resist it, unfortunately those mechanisms still have many limitations that are given below. The authors in [25] suggest disabling two main fields in NDN packet specification in order to mitigate the privacy of the content presented in the cache, such as for *"Paid-Content"*. Thus, by eliminating the scope field from the new packet specification, the *"hop-Limit"* field still can be manipulated as a scope field. As for the mechanisms presented in [26], [27] and [28], which propose applying an extra delay on the suspected interest with a specific prefix, this would delay the attack such as in the case of TBA, but it still can't resist if the intruder gets the desired content. In order to mitigate Cache Poisoning Attack accurately, [29] and [30] use a Self-certifying-based method. This method presents several drawbacks that may lead to false positive decisions. Based on this point, these methods are focusing more on static content, which means that the dynamic content can not be detected.

The above mentioned mechanisms suffers from the extreme exhaustion of the NDN router that can cause damage in the main components' functionality. In the case of IFA detection mechanism, the suggestion of [32] presents some limits. By relying on the edge NDN routers, the authors do not take in consideration that those routers may get effected by the malicious nodes. False positive decisions may appear as the legitimate consumers may request unavailable content. The authors of [34] base their solution on a neural network-based detection system, that may cause a huge space exhaustion and overload CPU usage. This mechanism keeps storing the data and the interest related information even in a non attacking state.

6 Conclusion

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 number of new attacks have appeared that target NDN's main components. In this paper, we illustrated the potential threats that could affect the NDN architecture. Most of these vulnerabilities result from the availability of the data in the cache of intermediate routers. The most important of these attacks are the Cache Poisoning Attack, the Cache Privacy Attack and the Cache Pollution Attack. Moreover, we provide an overview about new detection mechanisms that have been presented in the literature, together with their limitations.

Table 2. Detection mechanisms limitations

	Ref	Limitations								
		Compromisable	Identity leakage	Bandwidth usage	Space storage	CPU Overload	Accuracy	Topology	Complexity	False positive
CPA	[15]			✓	✓	✓			✓	
	[17]									✓
	[16]			✓	✓	✓	✓			
	[19]	✓								✓
	[22]	✓		✓	✓	✓				✓
Cache Privacy Attack	[25]	✓	✓				✓			
	[26]		✓					✓		✓
	[27]	✓					✓			
	[28]	✓								
Cache Poisoning Attack	[29]	✓								✓
	[30]	✓				✓				✓
IFA	[32]	✓								✓
	[34]			✓	✓	✓	✓			✓

References

1. Tchernykh, A., Babenko, M., Chervyakov, N., Miranda-Lopez, V., Avetisyan, A., Drozdov, A. Y., Rivera-Rodriguez, R., Radchenko, G., 38; Du, Z. (2020). Scalable data storage design for nonstationary iot environment with adaptive security and reliability. IEEE Internet of Things Journal, 7(10), 10171–10188. https://doi.org/10.1109/jiot.2020.2981276
2. Conti, M., Gangwal, A., Hassan, M., Lal, C., 38; Losiouk, E. (2020). The road ahead for networking: A survey on ICN-IP coexistence solutions. IEEE Communications Surveys 38; Tutorials, 22(3), 2104–2129. https://doi.org/10.1109/comst.2020.2994526
3. Deborah Estrin, Lixia Zhang and Jeffrey Burke, "Named Data Networking (NDN) Project", Technical Report, October 2010.
4. Mejri, S., Touati, H., Kamoun F.: Hop-by-hop interest rate notification and adjustment in named data networks. In: 2018 IEEE Wireless Communications and Networking Conference (WCNC), pp. 1-6(2018).
5. 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.
6. Kardi, A., Touati, H.: NDVN : Named Data for Vehicular Networking. International Journal of Engineering Research Technology (IJERT), 4(4), (2015).
7. Touati, H., Aboud, A., Hnich, B.: Named Data Networking-based communication model for Internet of Things using energy aware forwarding strategy and smart sleep mode. Concurrency and Computation: Practice and Experience. vol. 34, no. 3, 2022. https://doi.org/10.1002/cpe.6584

8. Mejri, S., Touati, H., Kamoun, F. (2016). Preventing unnecessary interests retransmission in named data networking. 2016 International Symposium on Networks, Computers and Communications (ISNCC). https://doi.org/10.1109/isncc.2016.7746058
9. NVD - CVE-2020-25681. https://nvd.nist.gov/vuln/detail/CVE-2020-25681. Accessed 17 Mar. 2022.
10. L. Deng, Y. Gao, Y. Chen, and A. Kuzmanovic, "Pollution attacks and defenses for Internet caching systems," Computer Networks, vol. 52, no. 5, pp. 935–956, Apr. 2008, doi: 10.1016/j.comnet.2007.11.019.
11. Kumar, N., Singh, A. K., Aleem, A., 38; Srivastava, S. (2019). Security attacks in named data networking: A review and research directions. Journal of Computer Science and Technology, 34(6), 1319–1350. https://doi.org/10.1007/s11390-019-1978-9
12. Hidouri, A., Hadded, M., Hajlaoui, N., Touati, H., 38; Muhlethaler, P. (2021, September 23). Cache pollution attacks in the NDN architecture: Impact and analysis. 2021 International Conference on Software, Telecommunications and Computer Networks (SoftCOM). http://dx.doi.org/10.23919/softcom52868.2021.9559049
13. 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-attackstr-iseclab-0812-001.pdf
14. Buragohain, M., 38; Nandi, S. (2020). Demystifying security on NDN: A survey of existing attacks and open research challenges. In The "Essence" of Network Security: An End-to-End Panorama (pp. 241–261). Springer Singapore. http://dx.doi.org/10.1007/978-981-15-9317-8_10
15. Mengjun Xie, Widjaja, I., 38; Haining Wang. (2012, March). Enhancing cache robustness for content-centric networking. 2012 Proceedings IEEE INFOCOM. http://dx.doi.org/10.1109/infcom.2012.6195632
16. Guo, H., Wang, X., Chang, K., 38; Tian, Y. (2016). Exploiting path diversity for thwarting pollution attacks in named data networking. IEEE Transactions on Information Forensics and Security, 11(9), 2077–2090. https://doi.org/10.1109/tifs.2016.2574307
17. Kamimoto, T., Mori, K., Umeda, S., Ohata, Y., 38; Shigeno, H. (2016, January). Cache protection method based on prefix hierarchy for content-oriented network. 2016 13th IEEE Annual Consumer Communications 38; Networking Conference (CCNC). http://dx.doi.org/10.1109/ccnc.2016.7444816
18. Zhang, G., Liu, J., Chnag, X., 38; Chen, Z. (2017). Combining popularity and locality to enhance in-network caching performance and mitigate pollution attacks in content-centric networking. IEEE Access, 5, 19012–19022. https://doi.org/10.1109/access.2017.2754058
19. Yao, L., Fan, Z., Deng, J., Fan, X., 38; Wu, G. (2020). Detection and defense of cache pollution attacks using clustering in named data networks. IEEE Transactions on Dependable and Secure Computing, 17(6), 1310–1321. https://doi.org/10.1109/tdsc.2018.2876257
20. Hidouri, A., Touati, H., Hadded, M., Hajlaoui, N., 38; Muhlethaler, P. (2022). A detection mechanism for cache pollution attack in named data network architecture. In Advanced Information Networking and Applications (pp. 435–446). Springer International Publishing. http://dx.doi.org/10.1007/978-3-030-99584-3_38

21. Park, H., Widjaja, I., 38; Lee, H. (2012, June). Detection of cache pollution attacks using randomness checks. 2012 IEEE International Conference on Communications (ICC). http://dx.doi.org/10.1109/icc.2012.6363885
22. Karami, A., 38; Guerrero-Zapata, M. (2015). An ANFIS-based cache replacement method for mitigating cache pollution attacks in Named Data Networking. Computer Networks, 80, 51–65. https://doi.org/10.1016/j.comnet.2015.01.020
23. Kumar, N., 38; Srivast, S. (2021). IBPC: An Approach for Mitigation of Cache Pollution Attack in NDN using Interface-Based Popularity. Research Square Platform LLC. http://dx.doi.org/10.21203/rs.3.rs-682924/v1
24. Lei, K., Fang, J., Zhang, Q., Lou, J., Du, M., Huang, J., Wang, J., 38; Xu, K. (2020). Blockchain-Based cache poisoning security protection and privacy-aware access control in NDN vehicular edge computing networks. Journal of Grid Computing, 18(4), 593–613. https://doi.org/10.1007/s10723-020-09531-1
25. Lauinger T, Laoutaris N, Rodriguez P, Strufe T, Biersack E, Kirda E. Privacy risks in Named Data Networking: What is the cost of performance? ACM SIGCOMM Computer Communication Review, 2012, 42(5): 54-57.
26. Ntuli, N., 38; Han, S. (2012, October). Detecting router cache snooping in Named Data Networking. 2012 International Conference on ICT Convergence (ICTC). http://dx.doi.org/10.1109/ictc.2012.6387155
27. Gao, M., Zhu, X., 38; Su, Y. (2015, November). Protecting router cache privacy in named data networking. 2015 IEEE/CIC International Conference on Communications in China (ICCC). http://dx.doi.org/10.1109/iccchina.2015.7448754
28. Kumar, N., Singh, A. K., 38; Srivastava, S. (2018). A triggered delay-based approach against cache privacy attack in NDN. International Journal of Networked and Distributed Computing, 6(3), 174. https://doi.org/10.2991/ijndc.2018.6.3.5
29. Gasti, P., Tsudik, G., Uzun, E., Zhang, L. (2013, July). DoS and DDoS in Named Data Networking. 2013 22nd International Conference on Computer Communication and Networks (ICCCN). http://dx.doi.org/10.1109/icccn.2013.6614127
30. Kim, D., Nam, S., Bi, J., Yeom, I. (2015, September 30). Efficient content verification in named data networking. Proceedings of the 2nd ACM Conference on Information-Centric Networking. http://dx.doi.org/10.1145/2810156.2810165
31. Ghali, C., Tsudik, G., Uzun, E. (2014). Needle in a Haystack: Mitigating content poisoning in named-data networking. Proceedings 2014 Workshop on Security of Emerging Networking Technologies. http://dx.doi.org/10.14722/sent.2014.23014
32. Dai, H., Wang, Y., Fan, J., 38; Liu, B. (2013, April). Mitigate DDoS attacks in NDN by interest traceback. 2013 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS). http://dx.doi.org/10.1109/infcomw.2013.6970722
33. Compagno, A., Conti, M., Gasti, P., 38; Tsudik, G. (2013, October). Poseidon: Mitigating interest flooding DDoS attacks in Named Data Networking. 38th Annual IEEE Conference on Local Computer Networks. http://dx.doi.org/10.1109/lcn.2013.6761300
34. Karami, A., 38; Guerrero-Zapata, M. (2015). A hybrid multiobjective RBF-PSO method for mitigating DoS attacks in Named Data Networking. Neurocomputing, 151, 1262–1282. https://doi.org/10.1016/j.neucom.2014.11.003