Evaluating and Mitigating a Collusive Version of the Interest Flooding Attack in NDN

Hani Salah
Computer Science Department, TU Darmstadt
Hochschul Str. 10, 64289 Darmstadt, Germany
hsalah@cs.tu-darmstadt.de

Thorsten Strufe
Computer Science Department, TU Dresden
Mommsen Str. 8, 01187 Dresden, Germany
thorsten.strufe@tu-dresden.de

***Abstract*—Named-Data Networking (NDN) is a promising architecture for the future Internet. However, it is hampered by interest flooding, an NDN-tailored DDoS attack which has been shown to cause dropping majority of legitimate packets. While several defence mechanisms have been suggested against it, they cannot protect NDN against the *Collusive Interest Flooding Attack (CIFA)*, a previously disregarded version of interest flooding. In CIFA, malicious clients issue interest packets that can be satisfied only by a malicious server. The server, in turn, responds with data packets just before expiration of the corresponding PIT entries.**

We study the effect of CIFA. Extensively simulating CIFA, we show that it affects the network and legitimate users almost as badly as an extensively researched version of interest flooding. Subsequently, we develop a generic defence mechanism against interest flooding attacks. The mechanism is based on CoMon, our framework for coordination in NDN. Thanks to CoMon, the attacks are detected and mitigated at an early stage by only a few routers. Via realistic simulations, we show that our defence decreases the amount of dropped legitimate packets remarkably, incurring a very low signalling overhead.

***Index Terms*—NDN; Collusion Attack; Interest Flooding**

I. Introduction

The Internet was designed in the 1960s as a network for connecting hosts in a reliable way. The same network, however, is used today mainly for another purpose: distribution and retrieval of content. Such usage (over the current Internet) is causing enormous and ever increasing traffic magnitudes [1]. This traffic, mainly attributed to redistribution of popular content, causes high charges for network operators and is mostly mitigated by costly content delivery networks (CDNs).

Several technologies have been proposed in the last years to match the Internet design and its usage [2]. Among them, Named-Data Networking (NDN) [3] is vastly looked at as a potential architecture for the future Internet. In essence, NDN replaces the current sender-driven host-centric communication model by a receiver-driven content-centric one.

Our work is motivated by the importance of treating security problems in a potential future Internet architecture before it is deployed in reality. We concentrate on interest flooding, an NDN-specific distributed denial-of-service (DDoS) attack which can result in dropping majority of legitimate packets [4]. Interest flooding misapplies two features of NDN: (i) routing based on longest name-prefix match and (ii) storing a forwarding state per interest packet in the routers' pending interest tables (PITs). Previous studies (e.g. see [4]–[9] and the references therein) focused on the *Non-Collusive Interest Flooding Attack (NCIFA)*, in which adversaries flood the network with *non-satisfiable* interest packets.

Instead, we study a formerly disregarded version of interest flooding which we coin with the name *Collusive Interest Flooding Attack (CIFA)*. In CIFA, malicious clients and a malicious server (i.e. content provider) collude together to achieve interest flooding. More precisely, malicious clients issue a large number of unique interest packets requesting contents that can be satisfied only by the malicious server, resulting in one PIT entry per interest packet in each NDN router on the path. The malicious server successively answers with data packets just before the corresponding PIT entries expire. Once some PITs on the path are overloaded, legitimate interest packets are dropped. Due to this unique design, CIFA cannot be mitigated by the defence mechanisms previously proposed against NCIFA.

Our contribution in this paper is twofold: First, we perform an extensive simulation study to analyse the drastic effect of CIFA both on the network resources as well as on satisfaction of the legitimate users. Second, we propose and evaluate a defence mechanism against CIFA (which also works against NCIFA). In particular, we adapt CoMon (our framework for coordination in NDN) for this purpose, motivated by two success stories: (i) coordinating caching-related decisions [10] and (ii) defending against NCIFA [4]. In both studies, CoMon was able to realize efficient and feasible coordination. With CoMon, attacks are detected and mitigated in a coordinated way, based on aggregated and timely knowledge of forwarding states. In more details, attacks are detected and mitigated at an early stage by a small number of strategically positioned routers, with the aid of a domain controller, based on continuous monitoring of PIT entries. Extensively simulating our solution, we show that it is effective (against both CIFA and NCIFA) and incurs a very low signalling overhead.

The remainder of this paper is structured as follows: We give an overview of NDN and interest flooding in Section II and review the related work in Section III. Then, we describe our defence mechanism in Section IV. In Section V, we evaluate both the effect of CIFA (and compare it to NCIFA) as well as the effectiveness and the signalling overhead of our defence mechanism. Finally, we conclude the paper in Section VI.

II. Background

In this section, we give an overview of NDN (Subsection II-A) and interest flooding attacks (Subsection II-B).

A. Named-Data Networking

Named-Data Networking (NDN) [3] is a potential architecture for the future Internet. It was initiated at Xerox PARC with the goal to match the host-centric design of the Internet with its current content-centric usage.

Design notions: NDN is based on four design notions:

1) *Networking named content:* Clients access content by its name, rather than locations or host addresses. For this, each content is identified by a unique hierarchical name (e.g. ***"/unime.it/iscc2016/papers/salah_cifa.pdf"***).
2) *Client-driven communication model:* The model is based on two packet types: *interest* and *data* packets. Clients use interest packets to request named contents. The content itself is delivered inside a *data* packet on the same path through which it was requested, but in the reverse way.
3) *In-network, on-path caching:* When a data packet is being delivered, a copy of it is cached in each router along the path between the server and the client. A content replacement algorithm (e.g. LRU or LFU) is used to replace data packets, in case the cache store is full.
4) *Content-based security:* Each data packet contains a digital signature (or a reference to it). The origin content provider calculates the signature over the content's name and the content itself, thus binding them with each other. The data packet contains information through which the creator's public key can be retrieved. This way, the authenticity and integrity of data packets can be verified no matter where they are retrieved from.

Router model: The router in NDN has three components:

1) *Pending Interest Table (PIT):* It stores content names of interest packets that are received recently but not satisfied yet. Each PIT entry also identifies the incoming interface(s) through which the corresponding interest packet was received. The router removes a PIT entry either when the corresponding data packet is received, or when the entry times out.
2) *Forwarding Information Base (FIB):* Acting as a routing table, FIB maintains a list of potential outgoing interfaces for different content names and name-prefixes.
3) *Content Store (CS):* It temporarily stores data packets passing through the router.

Handling interest packets: When receiving an interest packet, the router looks for a matching name in its CS. If the name is found, the router forwards the matching data packet to the same interface from which the interest packet was received. Otherwise, the router looks for the name in its PIT. If a matching entry is found but the interface from which the interest packet was received is not listed, the new interface is appended to the same entry, and nothing otherwise. This way, NDN routers avoid forwarding duplicate copies of identical interest packets. If no matching PIT entry is found, a new one is created, and then the FIB is consulted and the packet is forwarded accordingly.

Handling data packets: Upon receiving a data packet, the router first looks for the content name in its PIT. If found, the data packet is cached in the CS, then forwarded to the listed interfaces, and lastly the corresponding PIT entry is deleted. If no matching PIT entry found, the packet is discarded.

The aforesaid features of NDN (stateful forwarding, routing and forwarding without host addresses, in-network caching, and content-based security) increase the network robustness against several types of traditional DDoS attacks [5]. In particular, bandwidth depletion, black-holing, prefix hijacking, and reflection attacks are eliminated or at least mitigated in NDN by design. Furthermore, NDN is not vulnerable to DNS cache poisoning and similar attacks because name resolution is not required in NDN.

B. Interest Flooding Attacks

As mentioned above, NDN provides a built-in protection against several traditional DDoS attacks. Nevertheless, NDN is vulnerable to new types of DDoS attacks [11]. Interest Flooding is one of those attacks which received a lot of attention from the research community in the last years. It misuses two design features of NDN: (i) routing according to longest name-prefix match and (ii) storing forwarding states of interest packets in routers' PITs. Such an attack aims at overwhelming routers' PITs and/or content providers, so they cannot handle legitimate interest packets.

Researchers so far focused on a consumer-driven version of interest flooding, which we call the *Non-Collusive Interest Flooding Attack (NCIFA)*. The adversary in NCIFA, targeting a specific name-space, produces a large number of *non-satisfiable* interest packets and inserts them into the network through distributed malicious clients (i.e. bots). The content name in each of those packets consists of the target name-space as a prefix appended by a unique fake suffix. This results in creating a PIT entry per fake interest packet in each crossed router. Since malicious interest packets are non-satisfiable, the corresponding PIT entries remain till they eventually expire.

Alternatively, the *Collusive Interest Flooding Attack (CIFA)* requires malicious clients as well as a malicious server to collude together to achieve interest flooding. More precisely, as illustrated in the example in Fig. 1, malicious interests issue many *satisfiable* interest packets named with unique content names starting with the name-space of the malicious server (e.g. ***"/malicious_domain/"***). This results in creating a single PIT entry per each malicious interest packet in each crossed router (like NCIFA). The server, in turn, responds with data packets shortly before the corresponding PIT entries expire.

In both NCIFA and CIFA, succeeding to overflow PITs of some routers leads to drop subsequent interest packets including those belonging to legitimate users.

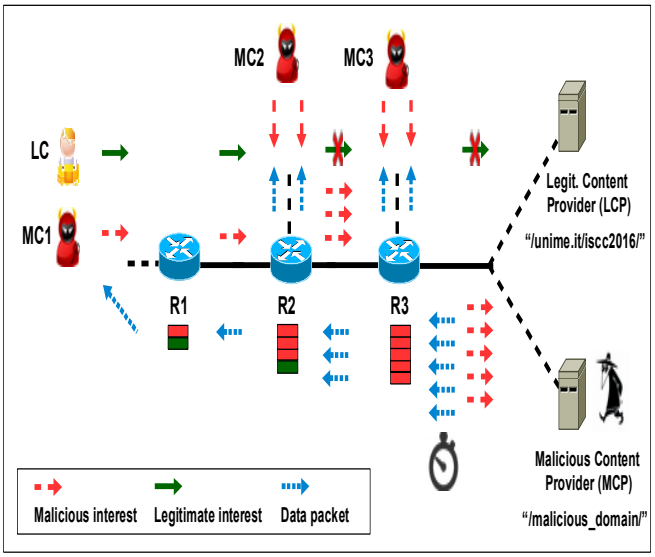


Fig. 1: Example of CIFA: There are three malicious clients (MC1, MC2, and MC3), one legitimate client (LC), one legitimate content provider (LCP) serving a legitimate name-space ***"/unime.it/iscc2016/"***, and one malicious content provider (MCP) serving a malicious name-space ***"/malicious_domain/"***. MC1, MC2, and MC3 produce (in total) five malicious interest packets requesting five different contents served by MCP. MCP responds with data packets just before the corresponding PIT entries expire. Since the moment the malicious interest packets were sent till the moment they are satisfied, LC attempts to request a content from LCP fail, because R3's PIT is full (assuming a maximum PIT capacity of five entries).

III. Related Work

Security issues of NDN and other information-centric architectures have been the subject of plenty of studies in the last years (see [11] for an overview). We focus in this section only on prior studies that are highly related to ours, i.e. those that dealt with interest flooding in NDN.

Interest flooding in NDN was discussed for the first time by Lauinger [12]. Subsequently, Gasti et al. [5] described three types of interest flooding. However, neither of the two studies evaluated the effectiveness of the attack nor they proposed defence mechanisms. After that, several studies (most notably [4], [6]–[9], [13]–[15], [15], [16]) analysed the effectiveness of interest flooding. Their evaluations showed that interest flooding is very harmful both for the network performance as well as for legitimate users. Some of those studies also proposed and evaluated defence mechanisms against interest flooding. Recently, Al-Sheikh et al. [17] classified some of them and compared them via simulations.

Despite the detailed analyses of interest flooding and the considerable number of proposed defence mechanisms in prior studies, CIFA was not considered. In [13], the authors only mentioned the idea of the attack. In addition, none of previous defence mechanisms (except [14], [15]) is directly applicable for CIFA. The authors in [14], [15] proposed to avoid interest flooding attacks by removing PIT. Although their solutions work against CIFA, they lose the performance and security gains that can be achieved with PIT [3], [18]. To our knowledge, we are the first who analyse CIFA and propose a defence mechanism against it (without removing the PIT).

Our defence mechanism is based on CoMon, our framework for coordination in NDN [4], [10]. More precisely, we use CoMon as an infrastructure for monitoring, attack detection, and mitigation. Although we employed CoMon for a similar purpose in [4] (initially in [19]), the original defence mechanism was designed for NCIFA. Due to the differences between CIFA and NCIFA (see Subsection II-B), the original defence mechanism does not work against CIFA. In this paper, we adapt the original defence mechanism so that it becomes more generic: it can defend against both CIFA and NCIFA effectively, while keeping the overhead low.

IV. Our Defence Mechanism

One simple solution to mitigate interest flooding attacks in NDN is to expand the maximum PIT size. However, we argue that such a solution should be avoided, because it will cause scalability issues and other drawbacks. In fact, several studies (e.g. Yuan and Crowley [20]) agreed on the necessity (and proposed solutions) to decrease PIT size.

Instead, we aim to achieve a defence mechanism that is scalable, effective, and lightweight (i.e. incurs low overhead). Lessons from [4] learned us that achieving an effective defence against DDoS in NDN requires: (i) to detect attacks accurately (even against low-rate attacks, and without duplication to avoid overreactions), and (ii) to detect and mitigate attacks at an early stage (before the attack causes a high damage).

While the intended benefits behind the last two requirements are charming, achieving them is challenging. They also contradict the requirement of low-overhead defence. More precisely, the two requirements presuppose that routers participating in the defence should have up-to-date global knowledge of attack-related information. The magnitude of such information on the level of the Internet or autonomous systems (i.e. domain networks) is massive, due to the large size of those networks, and also because this information should be exchanged very frequently (otherwise, obsolete attack-related information results in inaccurate defence-related decisions).

We propose to address this challenge by utilizing CoMon, our framework for **Co**ordination with lightweight **Mon**itoring [4], [10]. CoMon provides an infrastructure for network-wide coordination in NDN with low overhead. In [4], we utilized CoMon to defend against NCIFA (the non-collusive version of interest flooding). In that paper, we showed that our solution is highly effective against NCIFA and incurs a very low signalling overhead. However, the original attack detection and mitigation algorithms are not applicable for CIFA. Therefore, we need to adapt CoMon so that it can be utilized to effectively defend against both CIFA and NCIFA, while maintaining the overhead of coordination low.

In the remainder of this section, we first give an overview of the design primitives of the adapted version of CoMon (Subsection IV-A). Next, we detail the new algorithms of attack detection (Subsection IV-B) and mitigation (Subsection IV-C).

A. Design Primitives

System architecture: CoMon is designed to work within an autonomous system (i.e. domain network) consisting of a set V of routers. The system architecture, as illustrated in Fig. 2, consists of three principal components working together to defend against both CIFA and NCIFA, as follows:

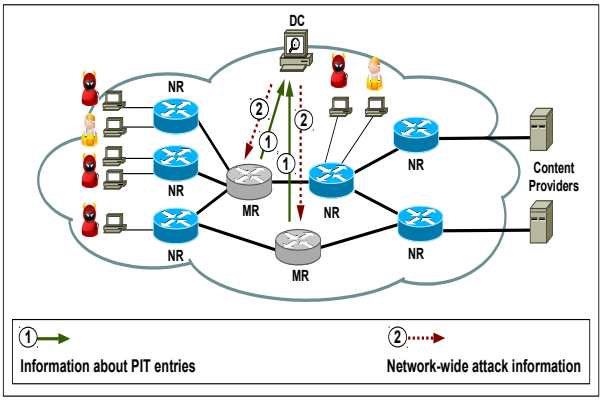


Fig. 2: System architecture (adapted from [4]): "DC" stands for Domain Controller, "NR" for NDN Router, and "MR" for Monitoring Router

1) *Domain Controller (DC):* Each domain network has a controller that collects monitored information about PIT entries from a predetermined subset of routers (hereafter, Monitoring Routers (MRs)). The DC uses this information to detect attacks. It then sends back (to the MRs) network-wide attack-related information.
   The DC is currently implemented in a centralized way which may cause scalability and single-point-of-failure issues. Those issues can be addressed by redesigning the DC in a distributed way. Such a design is out of the scope of this paper.
2) *NDN Routers (NRs):* These are similar to regular NDN routers [3] with a modified routing protocol.
3) *Monitoring Routers (MRs):* A set $M \subset V$ (where $|M| \ll |V|$) of routers are selected as MRs. They perform three additional tasks (not performed by NRs): (i) continuously monitor their PITs and send summaries of their observations to the DC, (ii) receive information of potential attacks detected at network-wide level (if there are any) from the DC, and (iii) use both their local observations and the information they received from the DC to detect and mitigate attacks.

To avoid duplicate monitoring (thus duplicate attack detection and overreaction), CoMon adds a field to interest packets called *"Checked"*. When the first MR on the path receives the interest packet, that MR sets the *Checked* field to 1 (0 by default). Only that MR reports about the corresponding PIT entry to the DC.

While adding new fields to the original packets in general is not desirable, the field that we add (i.e. *Checked*) neither significantly changes the structure of the packet (only one bit) nor violates NDN's design notions (Subsection II-A).

Monitoring techniques: Based on the discussion above, MRs should be: (i) jointly able to capture the entire network traffic (to defend against potential attacks based on network-wide related information), and (ii) located close to clients (so that attacks are detected and mitigated at an early stage). Towards this end, CoMon implements the following three techniques[1]:

1) *Placement of MRs:* CoMon implements a greedy algorithm called *PRCS* (Placement based on covered Routes and Closeness to Sources). PRCS selects a set of routers based on the number of routes on which they are located as well as their distances from the sources of interest packets. More precisely, for each route p of length $l(p)$, the algorithm weights each router r located on p as follows: $w(r,p) = 1 + \frac{P(r,p)}{l(p)}$, where $P(r,p)$ is r's position on p. In particular, $P(r,p) = 0$ for the gateway router (i.e. the closest to the target), and incremented by 1 for each hop towards the access router.
   Although PRCS enables for a high traffic coverage [4], it does not guarantee a full coverage. In addition, interest packets can be filtered by caches and PITs before they cross MRs. CoMon maximizes traffic coverage by implementing the following two techniques. When both are enabled, it is guaranteed that each interest packet (thus the corresponding data packet) crosses at least one MR.
2) *MR-Aware Routing (MAR):* CoMon implements a two-phase routing process on interest packets: (i) from the source to some (e.g. the closest) MR, and then (ii) from the designated MR to the original destination.
3) *Forward-Till-Be-Monitored (FTBM):* When a router receives an interest packet that is not monitored till that moment (i.e. $Checked = 0$) and finds a matching PIT entry or a matching data packet in its cache, the router only forwards the packet to the closest MR. The designated MR, in turn, considers the information of the packet for defence-related tasks, and drops it afterwards.

B. Attack Detection

Our defence mechanism detects attacks at two levels: (i) locally by each MR and (ii) globally by the DC. The mechanism exploits the consequences of interest flooding both to detect attacks as well as to mitigate them. More precisely, both CIFA and NCIFA result in a high PIT utilization. Consequently, our defence mechanism is mainly based on the *PIT utilization rate*. When the corresponding value is high, locally or globally, it is likely that an attack is ongoing.

Please note that *satisfaction of interest packets* (a widely used parameter for detecting NCIFA [4], [6], [7]) cannot be used with CIFA since malicious interest packets in CIFA are satisfiable.

Local attack detection: Each MR detects attacks locally by continuously monitoring its PIT and running Algorithm 1 at the end of each observation window q.

[1] The corresponding algorithms and overhead results (shown to be low) can be found in [4], [10].

Algorithm 1 Local attack detection

```
1:  L ← ∅                          ▷ Set of locally detected malicious name-prefixes
2:  Calculate the router's average PIT utilization rate U(q)
3:  for each name-prefix j do
4:      Calculate the average utilization rate U(j, q)
5:      Report U(j, q) to the DC
6:  end for
7:  if U(q) > α then
8:      for each name-prefix j do
9:          if U(j, q) > β then
10:             L ← L ∪ {j}
11:         end if
12:     end for
13:     if L ≠ ∅ then
14:         Trigger the mitigation function (locally)
15:     end if
16: end if
```

In Algorithm 1, the MR calculates the average PIT utilization rate at two levels: (i) over the entire PIT ($U(q)$, line: 2) and (ii) per name-prefix j ($U(j,q)$, line: 4). To avoid duplicate detection, $U(j,q)$ is calculated only over PIT entries corresponding to interest packets that are not monitored by another MR. In addition, the MR reports $U(j,q)$ to the DC (line: 5), to be used in detecting attacks at network-wide level.

Next, the MR detects potential attacks locally as follows (lines: 7 – 16): If $U(q)$ exceeds some preset threshold $\alpha \in [0,1]$, the MR assumes that some attack is ongoing. If so, the MR identifies a name-prefix j as malicious if $U(j,q)$ exceeds a preset threshold $\beta \in [0,1]$.[2] In case the MR identifies at least one malicious name-prefix, it triggers the mitigation function.

The (in)accuracy of local attack detection in Algorithm 1 depends on the values assigned to the aforementioned two parameters. On the one hand, using small values may result in false positives. On the other hand, using large values renders the detection difficult (i.e. insensitive), particularly for distributed low-rate attacks. We solve this dilemma by: (i) assigning the two parameters moderate to high values, and (ii) detecting distributed low-rate attacks globally.

Global attack detection: The DC performs global detection (Algorithm 2) after receiving the monitoring reports from the MRs at the end of each observation window q.

In essence, Algorithm 2 works as follows: For each name-prefix j, the DC aggregates the monitoring information received from the MRs (line: 3) and then use it to calculate the corresponding global utilization rate $U_{global}(j,q)$ (line: 4). Please note that $U_{global}(j,q)$ represents an upper bound on the number of PIT entries corresponding to j in any router. This information enables for early detection of potential attacks.

In case $U_{global}(j,q)$ exceeds some threshold $\delta \in [0,1]$ (lines: 5 – 7), the DC adds j to the list G of globally detected malicious name-prefixes. At the end (lines: 9 – 11), in case G contains one or more malicious name-prefixes, the DC reports G to all MRs and triggers the mitigation function in all of them.

[2] Please note that checking only $U(j,q)$ may cause false positives (e.g. $U(j,q)$ can be high because many popular contents belong to j). This problem can be avoided in Algorithm 1 by also checking that the time spent by the corresponding PIT entries in the PIT (normalized by the PIT's timeout) exceeds some other threshold $\gamma \in [0,1]$.

Algorithm 2 Global attack detection

```
1:  G ← ∅                          ▷ Set of globally detected malicious name-prefixes
2:  for each name-prefix j do
3:      Aggregate U(j, q)
4:      Calculate U_global(j, q)
5:      if U_global(j, q) > δ then
6:          G ← G ∪ {j}
7:      end if
8:  end for
9:  if G ≠ ∅ then
10:     Report G to all MRs and trigger mitigation in all of them
11: end if
```

C. Attack Mitigation

Once potential attacks are detected (locally or globally), each MR (independently) attempts to mitigate them by running Algorithm 3 along the next observation window.

Algorithm 3 Reaction against potential attacks

```
1:  while receiving interest packets do
2:      for each interest packet I do
3:          if Checked = 0 AND (j ∈ L OR j ∈ G) then
4:              Reject I with probability: U(j, q)
5:          else
6:              Accept I
7:          end if
8:      end for
9:  end while
```

In Algorithm 3, the MR checks each incoming interest packet. The principal idea is to use $U(j,q)$ to determine the probability of accepting or rejecting incoming interest packets (line: 4). That is, the larger $U(j,q)$ the larger the probability to reject (i.e. drop) the interest packet (and vice versa).

However, the algorithm directly accepts (line: 6) the interest packets that are already checked by another MR (i.e. $Checked = 0$) to avoid overreactions. The same applies for interest packets that do not belong to malicious name-prefixes (i.e. $j \notin L : j \notin G$).

V. Evaluation

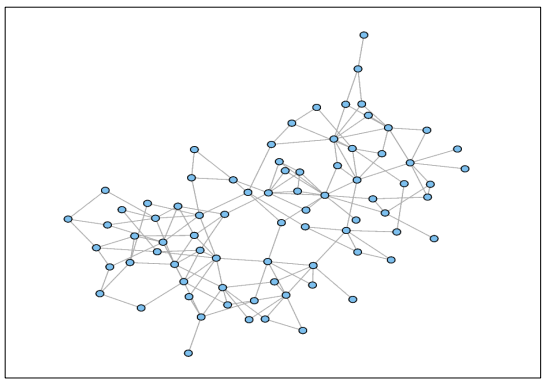
We performed an extensive simulation-based study to evaluate both: (i) the effect of CIFA on the network resources and legitimate users (compared with NCIFA), and (ii) the effectiveness and signalling overhead of our defence mechanism. We describe our evaluation setup and parameters in Subsection V-A and Subsection V-B, respectively. After that, we discuss the results in Subsection V-C.

A. Evaluation Setup

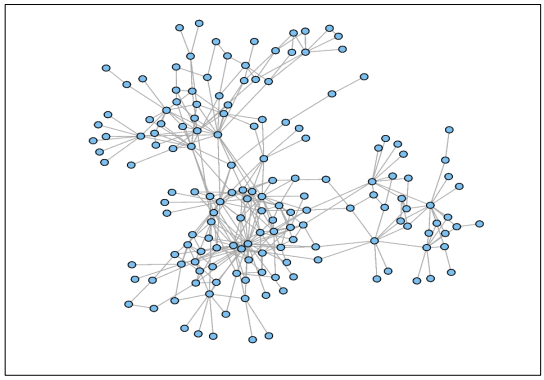
We implemented the attacks and our defence mechanism in ndnSIM [21], a widely used simulator in the NDN community. We fed the simulator with two real ISP network topologies

TABLE I: Basic properties of the two ISP network topologies used in simulations

Network	$\|V\|$	$\|E\|$ (bidir.)	Diameter	Avg. degree
AS-3967	79	147	10	3.72
AS-3257	161	328	10	4.08



(a) AS-3967



(b) AS-3257

Fig. 3: The two ISP network topologies used in simulations

measured by the Rocketfuel project [22] (Fig. 3): AS-3967 and AS-3257. We summarize their properties in Table I.

At the beginning of each experiment, the simulator selects 70% of the routers as access routers (through which clients connect to the network) and three of the rest as gateway routers (through which content providers are accessed). The simulator randomly selects 25% of the clients as malicious clients and one malicious content provider. Top $\lceil 10\% \rceil$ PRCS-ranked routers played the role of MRs.

Each legitimate client issued 100 interest packets per second (ipps). Malicious clients issued interest packets faster: we experimented with different rates ranging from 200 ipps to 5000 ipps. The effect of attacks slower than 200 ipps was small, while attacks starting at 2000 ipps or faster caused dropping large part of legitimate interest packets. In the following, we restrict our results discussion on two representative cases: 500 $ipps$ (low-rate attack) and 2000 ipps (high-rate attack).

We used the following configurations: PIT size of 5000 entries, PIT timeout period of 2 seconds, MR's observation window of 10 seconds, $\beta = 0.5$, $\alpha = \delta = 0.7$. In a real network, the last four values should be adjusted according to the network size as well as to the traffic volume. In order to not bias the signalling overhead results, we used a small data packet size of 1100 bytes.

Each experiment lasted for 540 simulation seconds. The attacks lasted between second 61 and second 360. This period was sufficient to evaluate both the effect of the attack and the effectiveness of the defence mechanism.

We repeated each experiment 20 times and obtained close results from repeated experiments. In the figures below, we plot the average, minimum, and maximum values. The values are computed over 600 data points: 20×30 data points per experiment (reported once every 10 simulation seconds along the attack period).

B. Evaluation Parameters

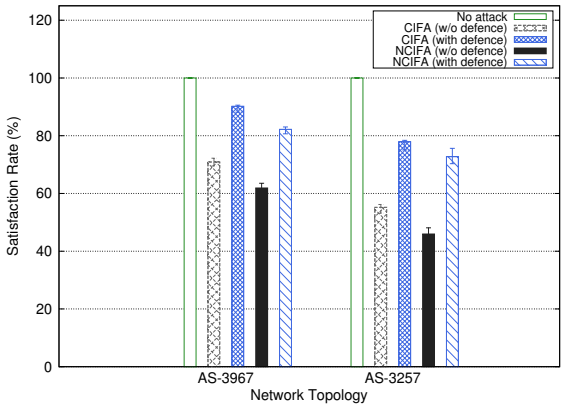
We use the following three metrics in our evaluation. The first two metrics measure the effect of the attacks as well as the effectiveness of our defence mechanism. The third one measures the signalling overhead of the defence mechanism.

1) *Satisfaction rate of legitimate interest packets:* This is a measure for the quality of experience of legitimate users during the attack period. An attack is considered effective if a low satisfaction rate is achieved. In contrast, a defence mechanism is considered effective when it significantly increases this rate.
2) *Network-wide PIT utilization rate:* This metric is also a measure for the effectiveness of both the attack and the defence mechanism on PIT (the targeted resource). An attack is considered effective if it increases the PIT utilization rate. In contrast, a defence mechanism is considered effective when it lowers this rate.
3) *Signalling overhead:* This parameter is measured by normalizing the total number of bytes used for defence (messages marked "1" and "2" in Fig. 2) by the total number of bytes of regular data packets.

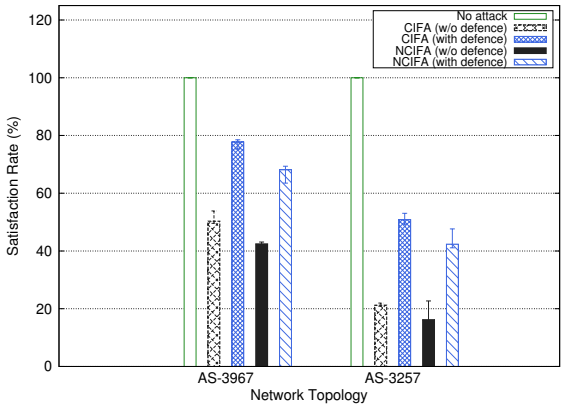
C. Results

Effectiveness of the attacks and the defence mechanism: In Fig. 4, we report the results of the satisfaction rate of legitimate interest packets under attacks both: (i) without defence and (ii) when our defence mechanism is enabled. Fig. 4a and Fig. 4b show the average, minimum, and maximum values under a low-rate attack and under a high-rate attack, respectively. Due to space constraints, we discuss the results of AS-3967 only. However, the same conclusions equally apply for the results of AS-3257.

Without defence, CIFA decreases the average satisfaction rate significantly: to about 70% under a low-rate attack and to about 50% under a high-rate attack. This means that 30% of legitimate interest packets under the low-rate attack and 50% of legitimate interest packets under the high-rate attack were dropped due to CIFA. In Fig. 4, we can also see that the effect of CIFA is only slightly lower (i.e. slightly less harmful) than NCIFA.

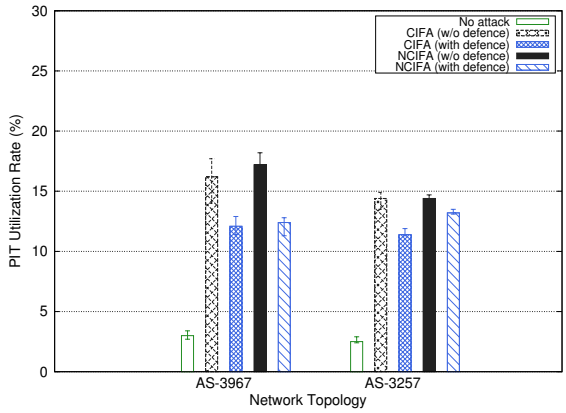


(a) Under a low-rate attack

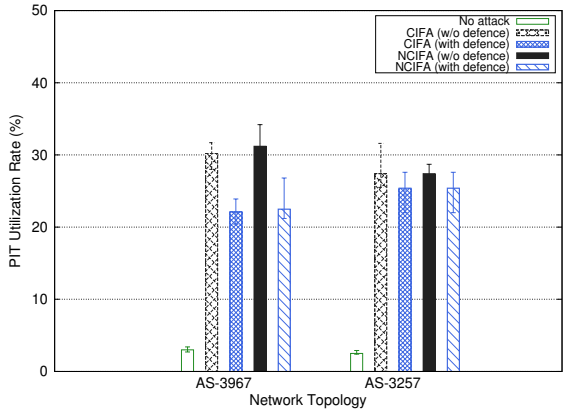


(b) Under a high-rate attack

Fig. 4: Satisfaction rate of legitimate interest packets



(a) Under a low-rate attack



(b) Under a high-rate attack

Fig. 5: Network-wide PIT utilization rate

Fig. 4 shows also that our defence mechanism is highly effective against both attacks. In particular, when our defence mechanism was enabled against CIFA, the satisfaction rates were improved by about 20% (from 70% to 90%) under a low-rate attack, and by about 28% (from 50% to 78%) under a high-rate attack. Similar improvements were achieved when the mechanism was enabled against NCIFA.

The results of the network-wide PIT utilization rate (Fig. 5) confirm both the negative impact of the attacks as well as the high effectiveness of our defence mechanism. That is to say, the attacks increase this rate while our mechanism decreases it when attacks are ongoing.

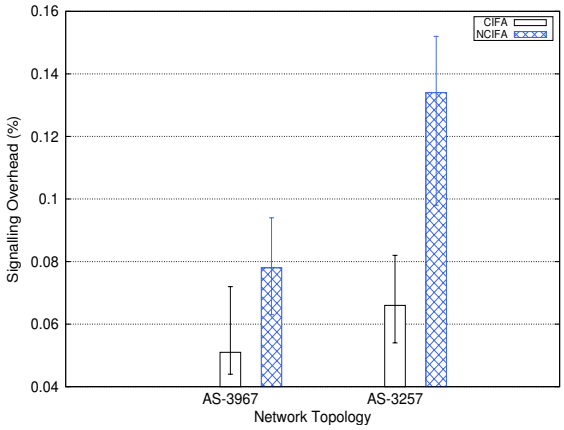
When our defence was disabled, CIFA increased the PIT utilization rate by about 13% (from 3% to 16%) under a low-rate attack, and by about 27% (from 3% to 30%) under a high-rate attack. However, our mechanism reduced the PIT utilization rate to about 12% under the low-rate attack, and to about 22% under the high-rate attack. Fig. 5 shows also that the results of NCIFA are similar to CIFA.

Please note that the improvement values which are reported by the network-wide PIT utilization rate are small (when compared with those that are reported by the satisfaction rate of legitimate interest packets). This can be explained as follows: only $0.25 \times 0.7 = 0.175$ of the routers can be used by the malicious clients in our setup. Those clients issue malicious interest packets towards a single content provider accessible through only one (out of three) gateway routers. It is thus highly probable that several routers are not located on the paths used by malicious interest packets, thus are not affected by the attack. Such routers, however, are counted when calculating the network-wide PIT utilization rate.

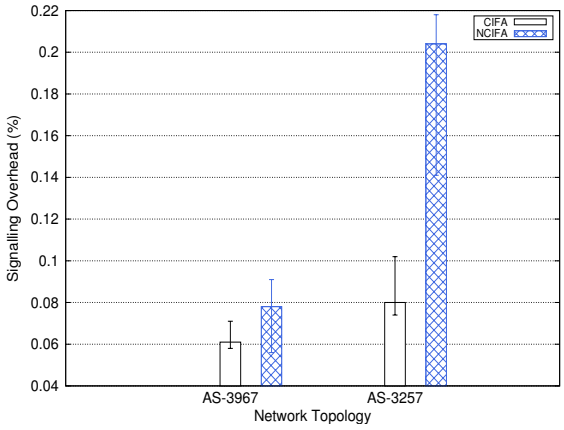
Signalling overhead: Finally, we report the signalling overhead that is incurred by our defence mechanism (Fig. 6). In Fig. 6a, we plot the overhead under a low-rate attack. When defending against CIFA, the maximum ranges from 0.072% in the AS-3967 topology to 0.082% in the AS-3257 topology.

In Fig. 6b, we plot the overhead under a high-rate attack. The maximum values here are higher than with the low-rate attack. In particular, the maximum ranges with CIFA from 0.071% in the AS-3967 topology to 0.102% in the AS-3257 topology. This raise is attributed to the increase in the number of messages exchanged between the MRs and the DC.

Under the two different attack rates (Fig. 6b and Fig. 6a), we can also see that the signalling overhead of our mechanism when defending against NCIFA is also very low. However, it is slightly higher than in the case of CIFA. This is because NCIFA is slightly more harmful, thus slightly more messages are expected to be exchanged between the MRs and the DC.



(a) Under a low-rate attack



(b) Under a high-rate attack

Fig. 6: Signalling overhead

The higher signalling overhead in the AS-3257 topology compared with the AS-3967 topology is to be expected. This is because the size of the AS-3257 topology (thus the number of MRs) is almost two times larger than the AS-3967 topology.

VI. Conclusion

We are the first (to the best of our knowledge) who analysed and mitigated a collusive version of the interest flooding attack (called CIFA) in NDN. Our evaluation results show that CIFA is very harmful both for the network and its users, resulting in dropping up to 79% of legitimate packets under a high-rate attack.

We consequently adapted CoMon [4], [10] (our framework for coordination in NDN) to defend against CIFA and another widely studied version of interest flooding. The results show that our defence mechanism is lightweight and highly effective against both attack versions.

In the future, we plan to improve our defence mechanism by (i) reimplementing the domain controller in a distributed way for fault tolerance and load balancing, and (ii) detecting and mitigating CIFA over multiple domains.

Acknowledgement

This work was supported by the Federal Ministry of Education and Research (BMBF), Germany, under the project "An Optic's Life" (no. 16KIS0025). We also thank the anonymous reviewers for their valuable comments, and Stefan Köpsell for the fruitful discussions.

References

[1] "Cisco Visual Networking Index: Forecast and Methodology, 2014–2019," *CISCO White paper*, 2015.
[2] G. Xylomenos *et al.*, "A survey of information-centric networking research," *IEEE Communication Magazine*, 2013.
[3] V. Jacobson *et al.*, "Networking named content," in *CoNEXT*, 2009.
[4] H. Salah, J. Wulfheide, and T. Strufe, "Coordination supports security: A new defence mechanism against interest flooding in NDN," in *IEEE LCN*, 2015.
[5] P. Gasti *et al.*, "DoS and DDoS in Named Data Networking," in *IEEE ICCCN*, 2013.
[6] A. Compagno *et al.*, "Poseidon: Mitigating interest flooding DDoS attacks in named data networking," in *IEEE LCN*, 2013.
[7] A. Afanasyev *et al.*, "Interest flooding attack and countermeasures in Named Data Networking," in *IEEE IFIP Networking*, 2013.
[8] D. Goergen *et al.*, "Security monitoring for content-centric networking," in *Springer data privacy management and autonomous spontaneous security*, 2013.
[9] H. Dai *et al.*, "Mitigate DDoS Attacks in NDN by Interest Traceback," in *INFOCOM Workshops*, 2013.
[10] H. Salah and T. Strufe, "CoMon: An Architecture for Coordinated Caching and Cache-Aware Routing in CCN," in *IEEE CCNC*, 2015.
[11] E. G. AbdAllah, H. S. Hassanein, and M. Zulkernine, "A Survey of Security Attacks in Information-Centric Networking," *IEEE Communications Surveys & Tutorials*.
[12] T. Lauinger, "Security & scalability of content-centric networking," *Master thesis, TU Darmstadt and Eurecom*, 2010.
[13] M. Wählisch, T. C. Schmidt, and M. Vahlenkamp, "Backscatter from the data plane–threats to stability and security in information-centric network infrastructure," *Elsevier Computer Networks*, 2013.
[14] C. Ghali *et al.*, "Living in a PIT-less World: A Case Against Stateful Forwarding in Content-Centric Networking," *arXiv preprint arXiv:1512.07755*, 2015.
[15] I. Widjaja, "Towards a flexible resource management system for content centric networking," in *IEEE ICC*, 2012.
[16] K. Wang *et al.*, "Cooperative-Filter: countering Interest flooding attacks in named data networking," *Springer Soft Computing*, 2014.
[17] S. Al-Sheikh, M. Wählisch, and T. C. Schmidt, "Revisiting Countermeasures Against NDN Interest Flooding," in *ACM ICN*, 2015.
[18] C. Yi *et al.*, "A case for stateful forwarding plane," *Computer Communications*, 2013.
[19] H. Salah, J. Wulfheide, and T. Strufe, "Lightweight Coordinated Defence Against Interest Flooding Attacks in NDN - (poster paper)," in *INFOCOM WKSHPS*, 2015.
[20] H. Yuan and P. Crowley, "Scalable pending interest table design: From principles to practice," in *IEEE INFOCOM*, 2014.
[21] A. Afanasyev *et al.*, "ndnSIM: NDN simulator for NS-3," *University of California, Los Angeles, Tech. Rep*, 2012.
[22] N. Spring, R. Mahajan, and D. Wetherall, "Measuring ISP topologies with Rocketfuel," in *SIGCOMM*, 2002.