Naming Content on the Network Layer: A Security Analysis of the Information-Centric Network Model

ELISA MANNES and CARLOS MAZIERO, Federal University of Paraná State, Brazil

The Information-Centric Network (ICN) paradigm is a future Internet approach aiming to tackle the Internet architectural problems and inefficiencies, by swapping the main entity of the network architecture from hosts to content items. In ICN, content names play a central role: Each content gets a unique name at the network layer, and this name is used for routing the content over the network. This paradigm change potentially enables a future Internet with better performance, reliability, scalability, and suitability for wireless and mobile communication. It also provides new intrinsic means to deal with some popular attacks on the Internet architecture, such as denial of service. However, this new paradigm also represents new challenges related to security that need to be addressed, to ensure its capability to support current and future Internet requirements. This article surveys and summarizes ongoing research concerning security aspects of ICNs, discussing vulnerabilities, attacks, and proposed solutions to mitigate them. We also discuss open challenges and propose future directions regarding research in ICN security.

CCS Concepts: • **Networks** → **Network security**; *Network design principles*; *Network layer protocols*;

Additional Key Words and Phrases: Information-centric network security, threat analysis, attack classification, countermeasures

ACM Reference format:
Elisa Mannes and Carlos Maziero. 2019. Naming Content on the Network Layer: A Security Analysis of the Information-Centric Network Model. *ACM Comput. Surv.* 52, 3, Article 44 (June 2019), 28 pages.
https://doi.org/10.1145/3311888

1 INTRODUCTION

The ubiquitous presence of the Internet points out its great success and emphasizes a future as promising as its past. A large portion of this success comes from the evolution of applications, services, and related technologies over the years, which have been shaping and evolving the way we use the Internet today. However, the Internet core architecture did not show the same evolution. Indeed, it has become extremely difficult to sustain the ever-increasing requirements for security, mobility, and availability through patches on protocols. This mismatch between the Internet architecture and its current requirements poses a huge motivation to look forward to a better

This work was supported by the Brazilian Coordination for the Improvement of Higher Education Personnel (CAPES) through a scholarship.
Authors' addresses: E. Mannes and C. Maziero, Federal University of Paraná State, Computer Science Dept. Curitiba, PR, 81531-980, Brazil; emails: elisa.mannes@udesc.br, maziero@inf.ufpr.br.

0360-0300/2019/06-ART44 $15.00
https://doi.org/10.1145/3311888

architecture, more dynamic, modular, and adaptive, suitable to accommodate services we experience today and foresee for the future [99].

The Information-Centric Network (ICN) paradigm [25, 53] has gained considerable attention for a future Internet from both academia and industry, precisely by overcoming current Internet shortcomings, by shifting the main network entity from hosts to content, thereby requesting and routing *named content* directly at the network layer. Basically, the ICN model defines content requests using application-layer names directly on the networking layer, as opposed to the current IP model, in which the networking layer deals with machine addresses. The ICN approach is supposed to rethink the Internet foundations and to design a nativeICN environment, embedding features like security, mobility support, and caching.

From the security perspective, the architecture shift imposed by the ICN paradigm on the network-layer changes many aspects related to network security. Naming content instead of machines changes the security paradigm from securing hosts and links to securing content, thus, naming content calls for authenticity and integrity protection mechanisms, to avoid well-known attacks such as spoofing. Moreover, naming content enables the detachment of a content from its location, allowing the deployment of in-network caches and resulting in content being retrieved from anywhere by anyone in the network. Although this brings obvious benefits, it also opens new challenges regarding privacy and access control on in-network cached content.

Furthermore, as routing and forwarding are performed by content names, ICN also raises security questions around monitoring and censorship. It becomes clear that ICN paves the way for potential security threats that are absent on current Internet; however, even known attacks from the Internet could undermine ICN architectures, as they may be emphasized when applied to ICN particularities. Thus, security in ICN should be provided differently from traditional networks. Consequently, security aspects in ICN need special attention, in order to ensure the architecture is robust to bear with current and future Internet requirements, and, more importantly, that the ICN paradigm is considered as a viable technology for content publishers.

This article aims to provide a broad view and insights regarding security threats and their countermeasures introduced by naming content directly on the network layer. The main contribution of this text is the systematic organization, classification, and discussion of ICN security issues and their possible solutions. It does not propose new solutions for specific attacks; instead, it discusses possible solutions and provides references to research works that detail them. We expect this article will open directions for a better understanding of current issues in ICN security and provide insights about new threats and potential directions for future works.

This study considers aspects of security threats in the three main ICN architectures: (i) Named Data Networking, Content-Centric Networking (NDN/CCN[1]), (ii) Network of Information (NetInf), and (iii) Publish-Subscribe Internet Technology (PURSUIT). The text reflects the amount of research on security in each of these ICN architectures. Since some of them, like NDN/CCN, are far more explored than the others, the amount of work discussed for each architecture is not evenly distributed. In consequence, the solutions discussed in this text are mostly related to the NDN/CCN architecture.

Some other surveys covering security aspects of ICNs were already published [1, 107]. Our text complements and updates them by analyzing, classifying, and organizing a broader number of papers about security in ICN and by giving an insight of current research and perspectives for

[1]CCN was the original ICN architecture as proposed by Jacobson et al. [54]. NDN is a more recent architecture [127], largely inspired by CCN and initially based on the same code base (formerly called CCNx and recently renamed to NDNx). In this text, we use the acronym NDN/CCN to refer to both architectures, as their characteristics relevant to this survey are equivalent.

future works. We also propose a brand new classification and organization for security attacks in ICN, enabling a new point of view on attacks and countermeasures in the ICN area. For a detailed review of the impact of some specific attacks in ICN architectures and the security requirements they impair, we refer the readers to AbdAllah et al. [1].

The remainder of this article is structured as follows: Section 2 introduces the main entities of an ICN and explains their basic operations (2.1), as well as presents the vulnerabilities and types of attackers we could expect in an ICN architecture (2.2). Then, we discuss threats, attacks, and countermeasures on main ICN aspects, structured into three main groups: security in content naming (Section 3), security in routers (Section 4), and security in caching (Section 5). Section 6 discusses challenges and opportunities on ICN security for the effective adoption of ICN architectures, as well as lists future research directions. Section 7 provides final remarks.

2 INFORMATION-CENTRIC NETWORKS

Since the Internet inception, we have seen a continuous change in the way we use it. Among the services offered on the Internet, content distribution currently stands out as the most used service. In fact, applications that generate real-time content, such as Netflix and Spotify, represent the category with the highest global monthly traffic, exceeding 50% in North America and with expectations of growing in the next years [99]. However, the Internet shows flaws in efficiently distributing content to respond to the increasing traffic volume, as a legacy from an architecture designed for resource sharing through point-to-point communication. Indeed, solutions to cope with this new content demand were proposed and deployed, such as multicast protocols, peer-to-peer (P2P) networks, and content distribution networks (CDN). However, such solutions are overlays on top of the traditional IP network, inheriting the limitations of such protocol.

The inefficiency of the Internet also extends toward other areas [93]: multiple copies of the same data being transmitted, leading costs to Internet Service Providers (ISPs); no native support for traffic optimization in the IP architecture; network congestion and high dissemination latency; security focus on connections rather than the content; connection-oriented protocols preventing mobility; weak management of energy, as more hardware are needed to perform redundant tasks, and poor end-user delivery performance, especially when it comes to streaming flows. Thus, the perspective of evolving the Internet architecture to one better suited for content distribution constitutes the main driving force behind the ICN paradigm. However, not only the shortcomings of Internet Protocol (IP) addressing are driving this change, but also the desire to embed the architecture with features to provide adaptability, security, mobility, self-organization, and natural evolvability, tackling some current problems and anticipating others.

2.1 The Information-Centric Network Paradigm

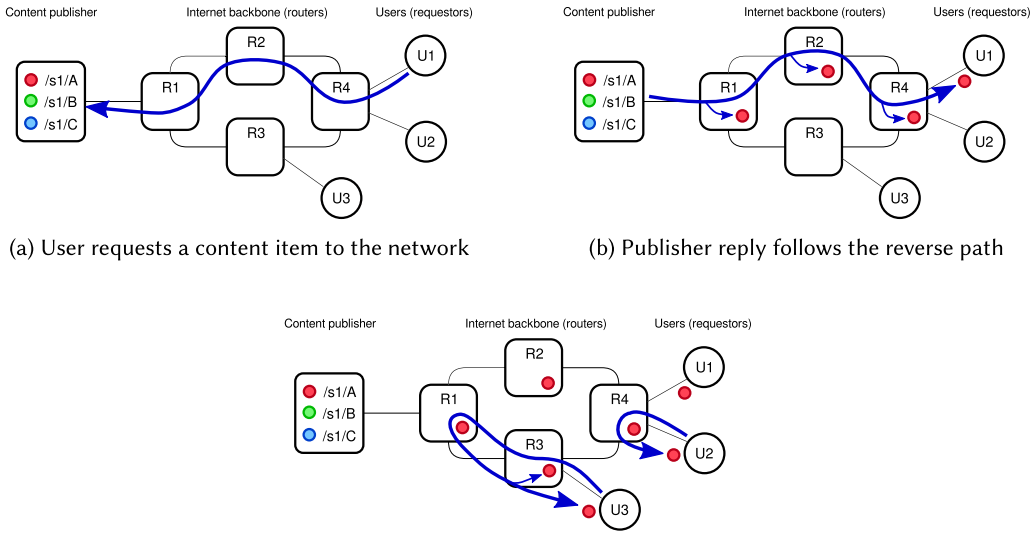
Comparing ICN to the IP model, Jacobson et al. [54] highlight two prominent differences: **what** the network routes and **how** it routes. In terms of what is being routed, IP routers traditionally route IP packets through the network to a destination machine, while ICN routers are supposed to route requests for a named content toward the best available copy. Thus, IP names machines while ICN names content. To route a request from the user to the destination machine, IP routers are sustained by a routing infrastructure (i.e., autonomous systems), and requests must follow the routing hierarchy toward the destination machine. Besides, network nodes are assisted by Domain Name Service (DNS) servers to translate human-readable names to IP addresses. The ICN paradigm also routes requests toward a content publisher; however, as it provides means of caching content in routers, these can satisfy the user without requiring the request to proceed all the way back to the publisher. In ICN, we observe three main domains that differ from traditional IP networking:

(i) naming, (ii) routing and forwarding, and (iii) caching. In the following, we highlight each of these domains.

- *Naming*: Since ICN names content items rather than hosts, there is a real concern about scalability. There are many more content items than hosts on the Internet, and the naming scheme is supposed to unambiguously identify all such content. Basically, naming schemes in ICN are divided into two major approaches: *flat naming* [29] and *hierarchical naming* [103]. Both have pros and cons: while flat naming allows better naming persistence, as it is not related to any location or organization, it also limits the aggregation of names for routing performance purposes. Hierarchical naming, in contrast, optimizes routing information aggregation and performance by a name prefix hierarchy, but groups of content that share the same prefix may have to be dealt together by the same location. It is also useful for naming content items in a particular context, facilitating management actions. There is also the *attribute-value naming scheme* [12], in which a set of attributes representing the capabilities of a content is mapped into a content name. Attribute-value naming schemes have been explored mostly on mobile and Internet of Things environments [90].
- *Routing and forwarding*: Two main forwarding schemes have being considered in ICN architectures: *name based routing* [54] and *name-based routing with support of a Name Resolution Service* (NRS) [4, 25]. Routers are populated with hierarchical prefixes and corresponding outgoing interface, and store the request state information to forward the content back to the requestor. Using this routing scheme, routers forward content based directly on the content name. In the routing scheme with support of a name resolution service, route discovery acts similarly to the traditional DNS: NRSs map content names to a set of locations; requests are then resolved by the NRS and forwarded by a topology based routing protocol, using the locations retrieved from NRS.
- *Caching*: three different caching solutions have been proposed to ICN architectures: *on-path*, *off-path*, and *peer-caching*. On-path caching opportunistically explores local content popularity in order to optimize content delivery. It caches content in routers based on the number of content requests passing through them. Off-path caching is performed by dedicated servers, very similar to CDNs today and is also sustained by content popularity. Peer-caching is tackled in mobile environments, where users' devices can be used as content caches to provide a better content availability to nearby devices. Caching also manages the eviction of content items through cache eviction policies such as Least Recently Used (LRU) and Least Frequently Used (LFU) [125].

To date, various ICN architectures have been proposed, such as NDN/CCN [54, 127], NetInf [25], PURSUIT [33], and Data-Oriented Network Architecture (DONA) [62]. Although these architectures share the same basic components (named content, routing by content name, and in-network caching), they have particularities regarding implementations. In this survey, we aim to systematize the security literature about the ICN core function vulnerabilities; thus, we do not discuss particularities of such architectures. Throughout this article, we refer to ICN principles and requirements as the *ICN paradigm*. The set of ICN instances we call *ICN architectures*. We thought this clarification would be valid as this distinction is not always clear.

Concerning ICN terminology, the RFC 7927 [64] defines three main terms: a *Named Data Object* (NDO), or simply a *data object*, is an addressable piece of information that can be requested from the network; a *Publisher* is an entity that publishes an NDO to the network (the publisher is not necessarily the NDO creator, it may just host data objects for the real producer); and a *Requestor* is an entity that issues a request for a named data object to the network. In this text, the terms *content* and *data object* are used interchangeably; the term *publisher* is used to represent both the



(a) User requests a content item to the network

(b) Publisher reply follows the reverse path

(c) Further requests are satisfied by network caches

Fig. 1. Basic ICN content distribution model.

entity that produces a content (its creator or producer) and the entity that makes it available to the network; the terms *requestor* and *user* are considered equivalent; finally, the term *source* designates a generic entity providing a data object, may it be a publisher or a cache.

Figure 1 depicts a simple example of how content flows in the ICN communication model, using the NDN/CCN architecture as a support for it. First, a publisher makes content available to users, in this example under the prefix `/s1/`, which became aware of such content by applications or search mechanisms. Users send requests for content items to the network (Figure 1(a)), without specifying a particular machine or address. The routers, based on their routing tables, route the request toward the content publisher, checking their caches for a cache hit. The reply containing the payload follows the reverse path (Figure 1(b)). As the content is forwarded through the routers, they may store a copy in their caches, providing ways to reason bandwidth, link quality, or available connections, by satisfying further requests without routing them all the way back to the publisher, as depicted in Figure 1(c).

2.2 Security Concerns in ICN

Despite the great benefits from adopting the ICN paradigm, the deep change it represents in the network layer invariably leads to new security challenges. For example, it is mandatory for all ICN architectures to provide a name-content integrity check mechanism, enabling users to check whether the retrieved content was tampered with. Furthermore, content authenticity should also be addressed, to provide means of assessing content origin. Content-based routing demands more management information to be handled in each router than IP-based routing, such as the Forward Information Base (FIB) and Pending Interest Table (PIT) structures in NDN/CCN. Since pending requests are stored by each router, they may be susceptible to Denial of Service (DOS) attacks by malicious users flooding them with requests [114] and should be protected. Caching mechanisms introduced by ICN may also be targeted by malicious users, and solutions to avoid well-known attacks as cache pollution and cache snooping need attention, as well as mechanisms to control access to cached content only to authorized users. Privacy is another concern for ICN architectures,

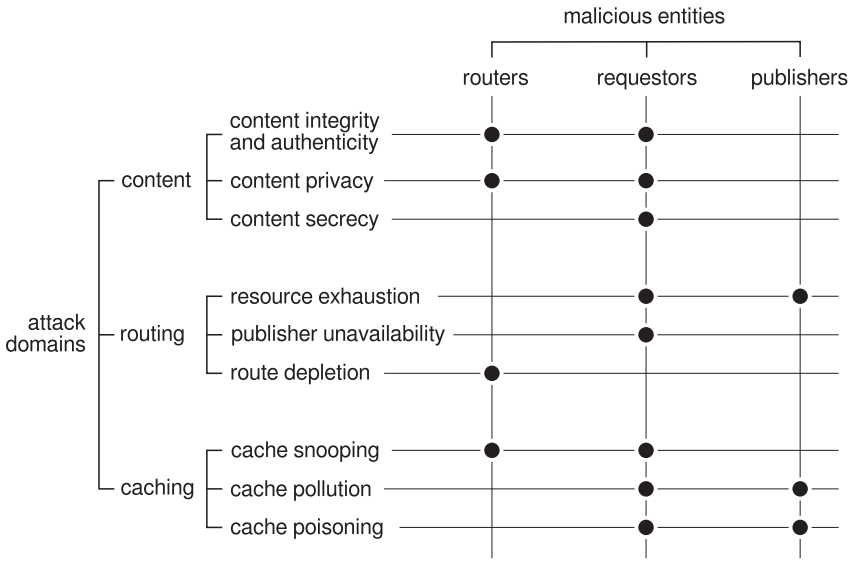


Fig. 2. Classification and organization of security threats and the malicious entities they involve.

as content items are requested by name and may be cached on intermediary routers; malicious entities should have no access to privacy-sensitive information through content names. ICN does not use the notion of *source address*; thus, it is not simple to bind a content to a requester, preserving her privacy. Nevertheless, the content name may reveal enough information to hinder users' privacy in some specific scenarios.

Overall, we consider that attacks exploring ICN vulnerabilities may be triggered by any entity participating in the network: users, content publishers, routers, and cache custodians. We denominate as ***malicious*** any of these entities that aims to explore vulnerabilities in the ICN architecture to disrupt the network or to jeopardize users' and content publishers' privacy. These malicious entities can manifest either in ***passive*** or ***active*** ways, depending on their interaction with the network. While passive attacks are difficult to detect, as the malicious entity does not interact with the network, active attacks can be profiled, as the malicious entity interacts with other network entities to issue an attack. Also, we assume that malicious entities can act ***alone*** or by ***colluding*** with others malicious entities to extend the damage. We also consider malicious entities that may have ***limited resources*** to launch attacks, as well as malicious entities that have ***high computational power*** available for attacks, such as routers, publishers, and governments.

To better understand how attacks influence the ICN behavior, we divided our analysis into the three main domains: security in content, in routers, and in caching. Figure 2 illustrates each of these domains and their respective classes of attacks, as well as the malicious entities that are most likely able to trigger them. For security in content, we expect attacks eased by the lack of security in content naming, leading to threats on content integrity, privacy, and unauthorized access. For routers security, we tackle attacks aiming to disrupt the network as a result of the ICN model, such as resource exhaustion, publisher unavailability, and route depletion. Finally, security in cache covers attacks to in-network caches as cache snooping, cache pollution, and cache poisoning. Each class of attack can be triggered by different vulnerabilities or security flaws in the ICN model, as detailed in next sections.

In the next sections, we systematically present and organize the main attacks against ICN discussed in the literature, and the countermeasures proposed to thwart them, when applicable. We

specifically target ***intentional faults*** introduced by attackers, although ICN architectures are also prone to other faults, like physical and design faults. We also introduce tables containing the compilation of ICN attacks and their corresponding category, target, vulnerability, as well as references to papers tackling countermeasures to mitigate them. In such tables, we aim to specify exactly the name of the attacks used by authors in their papers, even though some of them may have identical functioning. We chose to stick with the naming convention used by the authors, to help the readers to identify these similarities. These tables are divided into three parts, each one presenting a category of attacks, being placed at the end of the corresponding section. Although distinct ICN architectures [5, 25, 33, 53] have particularities regarding implementations, they share the same basic ICN components (named content, routing by content name, and in-network caching). Thus, we are interested in security issues concerning the core functions of the ICN paradigm, and will not discuss particularities of such architectures.

3 SECURITY IN CONTENT

The content naming scheme is at the core of any ICN architecture and thus is one of the most critical mechanisms to be secured. As important as the protection of the content name is the protection of the content itself. Mainly due to content-location independence introduced by the ICN paradigm, the content is susceptible to threats regarding integrity and authenticity, privacy, and secrecy. Indeed, name and content security has been a main issue for ICN designers, and a great amount of vulnerabilities and solutions have already been studied. In this section, we categorize content naming security issues into three groups: *content integrity and authenticity*, *content privacy*, and *content secrecy*. Content integrity and authenticity threats comprise attacks that aim to modify and tamper legitimate content payload, to cheat the network and the users, or to inject fake content in the network, as if it originated from a real publisher. Content privacy attacks aim to violate users' and content publishers' sensitive information by tampering communication channels (privacy issues due to caching are discussed in Section 5) and may result in censorship and user monitoring. Finally, secrecy issues comprise attacks that aim to give access to cached content items to unauthorized entities. These attacks and their countermeasures are detailed in the following.

3.1 Content Integrity and Authenticity

Because content may be distributed and cached by third-party entities, like routers and mobile devices, rather than just servers controlled by content publishers, it is subject to attacks targeting its integrity and authenticity [50, 97]. Malicious entities may tamper content items, by intercepting them and modifying their payload; they also can ***forge content*** and inject it in the network, as if it originated from a real publisher. Such attacks are simple and relatively easy to launch: first, the content name is visible to the network entities; thus, it is relatively easy for malicious entities to target popular content items. Second, because the content may be kept by any network entity, the content publisher has no control over it. However, it is easier for malicious routers to launch these attacks due to their privileged positions in the network. Figure 3 depicts this scenario; while user U2 requests for content `/s1/A`, router R2 produces and replies with a forged content. This attack is extremely damaging to users because it is expected that they accept content from locations other than the content publisher itself; thus, they are vulnerable to corrupted/suspicious content that does not correspond to the original request.

Concerns about content integrity and authenticity are fundamental for any ICN design and indeed have been considered since the conception of naming schemes by adopting ***digital signatures*** to provide guarantees on content integrity and provenance [45]. It should be noted that content integrity and authenticity are two closely related issues. Most solutions to content integrity threats also apply to content authenticity issues. In most of ICN architectures, the content itself

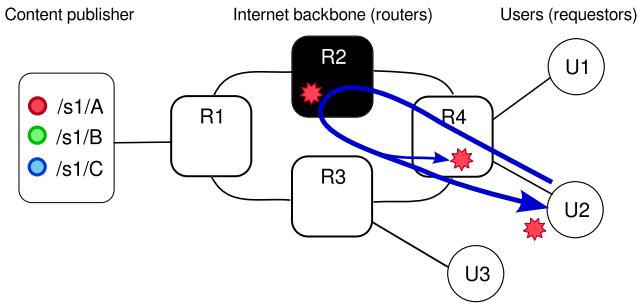


Fig. 3. Integrity threat in ICN.

can ensure its integrity and authenticity, thus allowing its location independence. For instance, the NDN/CCN architecture ensures integrity by ***authenticating the linkage between the content and its name***. This is performed by the content publisher, who digitally signs the mapping from the content name to the content [103].

Naming a data object after the ***cryptographic hash*** of its content allows checking the object's integrity. Thus, upon the reception of a content, the user is able to check its integrity and provenance by confronting content hashes and certificates with the content publisher or with support of metadata. This schema is adopted in architectures that use flat naming schemes, like NetInf [24], but can also be applied to hierarchical naming ones, as in NDN/CCN [13, 41]. One should observe that this approach works as long as the cryptographic hash function is trustworthy; a broken hash function in an architecture that uses hash-based name binding would allow one to easily publish distinct contents with the same name. Alternatively, ***identity-based encryption*** is also proposed to assess the integrity and authenticity of content items [48, 50, 109, 129], where the content name serves as a public key for the content publisher to validate its authenticity. However, such schemes still need the use of auxiliary tools that allow the verification of content integrity.

As essential as paying attention to content authenticity is addressing publisher authenticity, in which the user is able to identify and trust the content publisher. The user may decide to accept or reject a content based on trust information regarding the content publisher, thus involving ***trust management*** mechanisms [123]. As the users are able to assess integrity and authenticity from the content itself, they should also make sure the public key used to sign the content is trusted. Alternatives to traditional public-key infrastructures for content authentication are also proposed in ICN, mainly exploring distributed and decentralized mechanisms such as the use of ***distributed hash tables*** (DHT) [117, 118], ***social graphs*** [74], and ***traditional DNS-like mechanisms*** [75]. Another solution propose splitting public keys into several pieces that are redundantly scattered on the network; users can retrieve them from any entity to verify content integrity [55, 61] or even use ***edge routers to validate the authenticity*** of all content entering their networks [60]. Researchers also explored a scheme for authenticate old data by using a bookkeeping service that certificates data signatures anytime [124]. The main challenges in the design of trust management schemes are scalability and revocation of keys and certificates [82], since the keys and certificates can also be stored on uncontrolled caches.

3.2 Content Privacy

One of the main mechanisms of any ICN architecture is the name-based routing, which consequently makes content names visible to the network. This is a problem because names may contain semantic information about content items, thus hindering privacy practices. Malicious entities could explore this feature to ***monitor***, ***filter***, and ***block*** users' requests based on content names

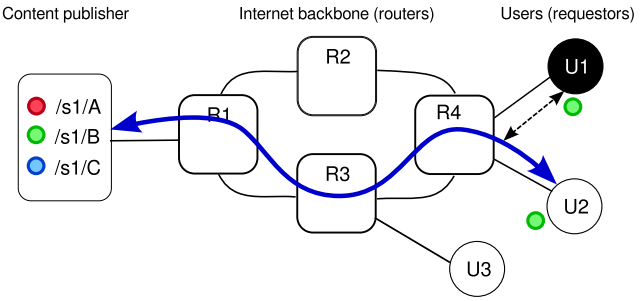


Fig. 4. Privacy threat in ICN.

[20]. For example, a malicious entity may use a blacklist with content names to block/delete and then monitor channels (and caches) to match content with the blacklist. As routers are strategically positioned in the network and have access to the network traffic, they are the best candidates to issue such attacks. Figure 4 illustrates U1 attempting against U2 privacy by monitoring the communication channel and learning what U2 is requesting. Alternatively, the malicious entity could simply inspect caches and channels for keywords [10]. Although it is difficult to pinpoint the specific user requesting the content, content monitoring and inspection could deny services or censure sensitive content items. This same tactic can be used for ***monitoring user requests***, in which the malicious user eavesdrops the communication channel to figure out content names and to infer users preferences [2].

From the privacy perspective, the main countermeasure to ensure privacy in ICN is to ***hide or mask the content requests*** from network entities [106]. ***Bloom filters*** [14] have been extensively explored in ICN for this purpose, since this scheme allows a network entity (e.g., a router) to test whether a particular content name is in the routing table or in the cache, without revealing the content name [16, 79]. Before requesting a content, the user computes the Bloom filter for each sub-component in the hierarchical name. Then, routers perform the longest-prefix matching on the content identifier. As this approach transforms the content identifier in a random string of bits, it helps to prevent blacklist matching and user profiling. Alternatively, ***homomorphic encryption*** can also be used to hide requests, at the same time that allows routers and content publishers to check whether the requested content is in their content sets, ensuring that neither content publishers nor third-party elements, such as eavesdroppers, are able to deduce or discover the content the user is requesting [33].

Another approach used to provide privacy to users when requesting content in ICN is by masking the content, using ***cover files***, for example. The idea of such solutions is to create a computational asymmetry for legitimate and malicious users when retrieving a content, avoiding blacklist attacks. Basically, the publisher mixes a legitimate content with a cover content. To retrieve the content, the user needs to gather some side information from the publisher using a secure channel, namely the content hash and the algorithm used to generate the covered content. As content items are retrieved in covered blocks, it requires a huge computational effort from malicious users to uncover the content being requested without prior knowledge of the extra information, while legitimate users could easily extract the desired content using the information retrieved in the secure side channel [10]. Another idea for the user wanting privacy in content retrieval is to encapsulate the request with each router's public key on a known circuit to the content, in an ***onion routing overlay***. Upon reception, each node on the circuit extracts its encryption layer and forwards the request to the next node. This proceeds until the request reaches the publisher, which will receive it in plain text [26, 101, 108]. Also, a user may encode the content name on requests and have

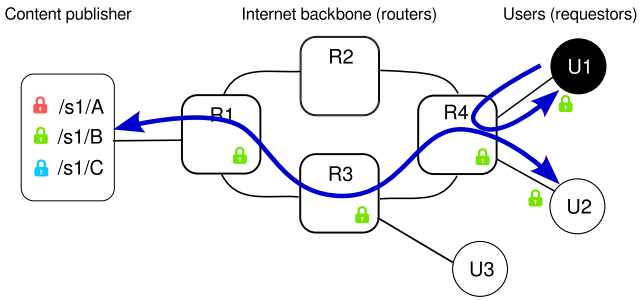


Fig. 5. Unauthorized access threat in ICN.

perfect secrecy, with the support of content publishers [106]. However, this solution generally impairs the benefits of in-network cache, negatively affecting network and caching performance.

Privacy is also a concern for content publishers due to the possibility of censorship, mainly because the content name may contain the publisher (producer) identification. Although censorship should not be handled by the network itself, the ICN behavior may ease it; thus, it is important to discuss such threats. A set of preventive measures to avoid monitoring and censorship of content publishers is proposed in Acs et al. [2], such as using ***group signatures, ring signatures, and confirmation signatures***, which allow a user to verify whether the signature of the content is valid, but not specifically identifying which publisher has signed it, and thus guaranteeing the content publisher's anonymity. Another idea is to use ***ephemeral identities*** under a reliable, permanent identity. The permanent identity is trusted by users; from there, different identities may be provided on demand to other content publishers, without revealing which one is signing the content.

3.3 Content Secrecy

The enforcement of access control policies to protect content secrecy in ICN is a great concern, especially due to the in-network caching infrastructure proposed by the ICN paradigm [52, 73]. As the retrieved content is cached along the way by unreliable entities such as routers, mobile devices, or third-party servers as in a CDN-like infrastructure, content publishers face problems to manage and enforce access control to their content. This problem is even more concerning when considering paid or copyrighted content, since cached copies of protected content may be accessed by users that do not have an account/subscription for it. Any user, despite her computational power, can retrieve unauthorized content from caches, since they do not validate access from users before replying. Figure 5 introduces a basic unauthorized access scenario, as U1 retrieves a protected content from router R4 cache, which was previously requested by the legitimate user U2. Restricting the name of the content only to authorized users is not enough, since routing and forwarding in ICN are carried out directly by the name of the content, therefore names can be easily discovered. Thus, this type of application requires a more robust and appropriate solution for use in ICN. Although access control problems are intrinsic to the ICN paradigm due to in-network caches, malicious entities may also benefit from traditional attacks used to gain access to protected content, such as Sybil attacks [84].

The basic action to maintain content secrecy in ICN is through ***content encryption***, ensuring that only users having a valid key can access it [17, 49, 53, 72, 116, 130]. However, depending on the encryption solution adopted, encrypted content may stumble upon caching versus access control trade-offs, mainly because content encrypted for a specific user using traditional symmetric encryption may not be cacheable to other users, who would not be able to decrypt it [76, 95]. To

circumvent that, special cryptographic mechanisms have being proposed for access control in ICN. ***Attribute-Based Encryption*** (ABE), for example, encrypts content for a group of users sharing common attributes, thus users are able to decrypt the content only if their key satisfies access control policies embedded in the ciphertext or in the key itself [52, 69, 70, 92]. In ABE schemes, the cached content can be shared among users in the same group. In the ***Broadcast Encryption*** (BE) model, content publishers encrypt the content with a unique symmetric key and distribute that symmetric key encrypted under the broadcast group public key. Each user in the broadcast group can decrypt the symmetric key using her individual private key. In the BE model, cached content can also be shared among users in the same broadcast group [49, 84, 94, 126]. However, if such solutions use the same key to encrypt content for a large group, key leakage may be a problem, as any user with the key can retrieve content, even unauthorized ones [78].

Another cryptographic scheme explored is ***Proxy Re-Encryption*** (PRE) [11]. In PRE-based solutions, content items are encrypted with a symmetric key. The symmetric key is encrypted with the content publisher's public key and, to recover the symmetric key, users must retrieve a re-encryption key with the content publisher [119]. A common problem of such solutions is the symmetric key disclosure. Once users have the symmetric key, they can access the cached content, even if they are not authorized. A variation of PRE solution is to encrypt each content with a distinct private key from an asymmetric key pair; thus, content items can be shared by all users through caches, while the content public key, if disclosed, is valid only in conjunction with the corresponding user's public key [77]. Different strategies, such as mixing symmetric and asymmetric encryption techniques [49, 76, 104, 126] and building access control frameworks that can be used in conjunction with any encryption-based solution [47, 63], have also been explored in the context of ICN. However, recent research discussed that encryption-based solutions are not sufficient for content secrecy, mainly because it is easy for attackers to infer a content by its popularity, even if they are encrypted [42].

Apart from encryption-based solutions, researchers also proposed alternative ***infrastructure-based solutions***, such as using authorization servers for validating policies [31, 32, 36, 102]. One problem of such solutions is that they often assume that content custodians (e.g., caches) validate access policies before sending the content to users. These assumptions may be hard to ensure on the Internet environment, for some reasons. Access control decisions involve evaluating rules defined on users and contents. For a cache to validate an access request, it needs to evaluate access control rules defined by the content publisher for that content and that user. This means that each cache should be provisioned (and updated) with the access rules for all contents it may store and all users it may serve, which is not scalable. Otherwise, caches may outsource the access decisions to authorization servers that know all rules, thus adding extra latency to every content request. The use of access lists is also considered [15, 44]; however, it is more suitable for controlled environments such as sensor networks at small places rather than the Internet. Another idea is to use levels of access directly on routers, such that routers are able to cache only content items which the publisher authorizes to [71]. Such solution assumes that the router operating system is modified to comply with access rules attached to content items and enforced by the router itself on forwarding and caching process. This implies caches should be able to evaluate and apply the access rules defined by the content publishers, which demands more processing for each request. Furthermore, this also imposes a common access control schema in the entire network for all publishers and caches.

Another face of access control is the use of ***firewalls*** considering ICN characteristics to support network administration. This can be used, for example, to enforce local security policies. Firewall rules are implemented in terms of blocking content by name or by publisher. If a certain publisher becomes malicious, rules could be implemented to avoid content from such publisher to reach the

Table 1. Overview and Classification of ICN Attacks and Countermeasures on Content

Attack	Category	Vulnerability	Countermeasures
Forging content	Content integrity and authenticity	Non-encrypted content	[13, 24, 45, 48, 50, 62, 75, 103, 109, 117, 118, 129]
Content renaming		Poor content-name binding/hash	
Content corruption		Lack of content encryption	
Name watchlist	Content privacy	Network routes named content	
Content analysis			
Name privacy			
Publisher privacy		Crypto keys identify the publisher	[2]
Unauthorized access	Content secrecy	Caching protected content	[15, 31, 32, 36, 44, 47, 49, 52, 63, 69–71, 77, 84, 92, 94, 95, 102, 119, 130]
Sybil (impersonation)		Credentials checking	[84]

network. Also, content could be checked against keywords for filtering purposes. Furthermore, as each content must be signed by its publisher, the firewall could be configured to ignore content with invalid signatures, for instance [44].

Table 1 presents the compilation of works addressing attacks and countermeasures on content security for ICN.

4 SECURITY IN ROUTERS

Routers perform essential services for the core of the network layer, as routing and forwarding. Besides managing and updating all routing information, it takes care of finding content in the network, forwarding requests and content back to the requestor.routing information. As the ICN paradigm brings new features to the network layer, new threats in routers functions also arise, mostly due to forwarding based by names directly in the network layer. However, even known attacks as denial of service may hamper ICN forwarding. We categorize routers attacks in ICN in three groups: *resource exhaustion*, *publisher unavailability,* and *route depletion*. Resource exhaustion attacks comprise malicious entities that demand routers unnecessary computations. Publisher unavailability groups attacks targeting content publishers to limit or stop the content distribution. At last, route depletion attacks aim to disrupt routing paths, e.g., by advertising invalid routes. In this section, we detail each of these groups.

4.1 Resource Exhaustion

To be able to forward the content back to the requestor, ICN architectures based on stateful routing, like NDN/CCN, demand routers to keep record of received and forwarded requests per interface, until the request is consumed, either by the content traversing the path back or by timeout expiration. This mechanism is pointed out as potentially vulnerable to Denial of Service (DoS) attacks aiming to disrupt forwarding service for legitimate users or to overload the network and disable it. The exploitation of this vulnerability has been extensively addressed in the literature, and can be effectively issued by any user in the network. For example, a malicious user or a set of colluding users could explore the stateful nature of routing in ICN by ***issuing an excessive number of requests***, consequently depleting the available memory for pending requests entries and denying

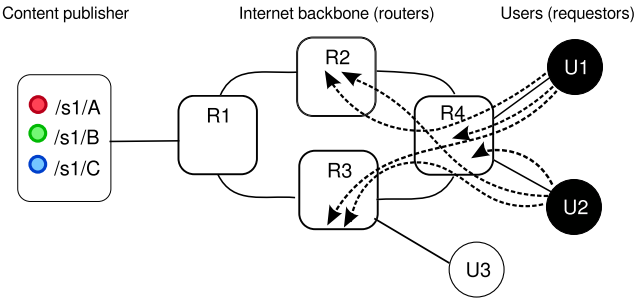


Fig. 6. Resource exhaustion threat in ICN.

service for legitimate users. Whether the router employs a rate limit for pending requests entries is indifferent, as it is simple for malicious users to exceed this limit [112]. However, since requests entries for the same content are aggregated for performance, this attack is ineffective or minimized if the malicious user sends a burst of requests for the same content, even in a distributed way. To disable this natural defense, malicious users should ***issue an excessive amount of bogus requests*** in order to fill the pending request table with forged entries, thus causing legitimate requests to be dropped [3, 18, 21, 23, 28, 34, 59, 110, 112]. Such attacks would be much more aggressive and difficult to detect with distributed attackers working simultaneously [43], as depicted in Figure 6. In this example, U1 and U2 together are exhausting resources from router R4, which may not be available to accept any more requests due to the amount of requests in its pending request records.

All these flooding attacks have a common consequence: They cause pending request entries to expire. To avoid this characteristic from being explored in countermeasures for flooding attacks, the user may ***collude with a malicious content publisher to avoid the detection of excessive requests being expired*** [65, 111, 114]. In this case, the malicious publisher replies forged requests right before the pending request entry timeout expires, causing the router to believe in a channel congestion. As a consequence, re-transmissions are triggered, amplifying the damage. These attacks can also be explored by two subverted routers slowing down content forwarding, forcing pending request entries in other routers to expire before content delivery is complete, thus causing re-transmissions storms [111].

The first attempts to mitigate flooding attacks in ICN suggest the ***use of hash functions on pending request tables*** to save storage or ***always accepting a new request***, dropping the oldest ones [65]. However, these are simple ideas that may delay attack consequences but not avoid them. More sophisticated solutions are based on monitoring and identifying unusual amounts of requests in routers [89]. For example, routers may ***monitor*** unsatisfied requests rate [7–9, 30, 34, 43, 56, 58, 59], the amount of entries on the pending request table [21], or the amount of requests on each interface [3, 88]. Thus, in case routers identify abnormal amounts of requests for distinct content items in the same interface, they can limit the amount of accepted requests from such interface, possibly minimizing the consequences of flooding while allowing legitimate requests from this interface to have a fair chance of being satisfied. Some solutions allow these results to be shared among routers, thus helping other routers to avoid or to recover from flooding attacks [43, 116].

Interest NACK (*non-acknowledgement*) packets can also be used to avoid routers to wait for malicious Interests timeout expiration. Using NACKs, routers can satisfy requests in routers, thus freeing PIT from malicious entries, if it is the case [122]. Also, routers may respond to suspicious requests to clean pending requests table on the way back to the malicious user, minimizing the impact of such attacks [23]. Another solution would be to adopt ***authenticated requests*** as a mean for routers to identify malicious users who issue an excessive amount of requests [15]. Also,

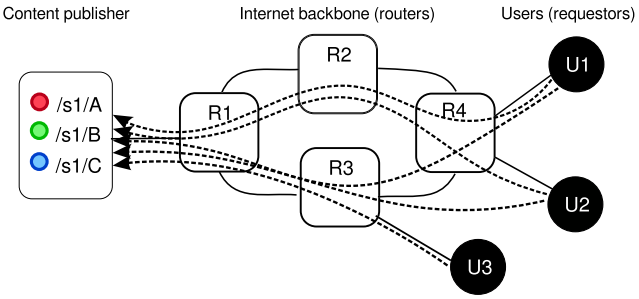


Fig. 7. Publisher unavailability threat in ICN.

adopting an ICN architecture based on stateless forwarding would decouple malicious entries from PIT [40, 115].

4.2 Publisher Unavailability

Just as sending excessive requests disrupts routing service for legitimate users, concentrating requests toward a unique publisher or a specific name space could also disable this entire name space or publisher, such as in the traditional IP network DoS attack. The most basic way of saturating the victim resources and making content unreachable for legitimate users is by ***sending a large amount of requests for the same content publisher***, in which malicious users collude to flood the network with requests toward the same victim server [34, 65], as illustrated in Figure 7. This attack requires a substantial amount of requests toward the same publisher to be effective; however, it does not require a significant amount of computational power from attackers, since the coordination of a group of users could be enough to deploy it.

However, targeting this kind of attack in an ICN infrastructure needs some special preparation by the malicious users. For example, they need to make sure that the content is not satisfied by any cache on the path, as well as that content requests are routed toward the victim and that each request creates new pending request table entries in routers. To subvert such adversities, malicious users could ***send bogus data requests with the same prefix***, inducing the routing mechanism to forward the request toward the prefix publisher, while preventing content to be found in cache [3, 21, 34, 113]. Due to the nature of these attacks, ***monitoring solutions*** used to mitigate flooding attacks can also be applied for these cases. Specifically against publisher unavailability attacks, routers could adopt metrics such as monitoring the amount of requests for the same content publishers, and reducing the requests forwarded to this prefix [65], avoiding that malicious requests overwhelm the content publisher.

4.3 Route Depletion

To correctly forward content requests toward publishers or available cached copies, each router needs to maintain its Forwarding Information Base (FIB) updated and free of malicious entries. Threats against FIB could potentially let legitimate domains unreachable or allow malicious users to redirect routes to malicious content or routers for monitoring purposes. This attack demands that attackers have access to route announcements; thus, it requires the subversion of routers or content publishers. Malicious content publishers could ***announce numerous malicious content belonging to different domains***, until the table is full and unable to register valid content from legitimate domains [43]. They could also ***announce content at a rate extrapolating the route update convergence time***, causing problems with incorrect information about cached content, misleading routes, and even opening security gaps for others attacks [111]. Alternatively, a

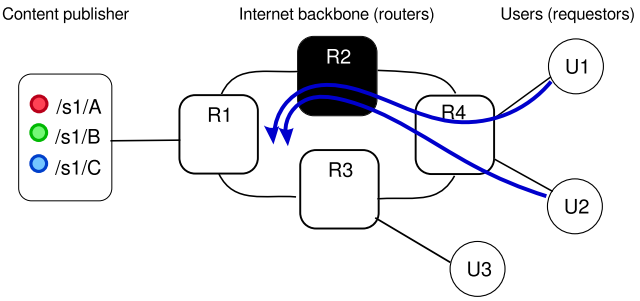


Fig. 8. Route depletion threat in ICN.

malicious router could ***announce valid content to the network and when asked for content retrieval, to delay the reply or even do not reply***, causing a disruption in services for users [34, 73, 112]. This attack is worsened if several malicious routers collude to announce routes to popular content and then delay replies [111]. Figure 8 depicts a scenario where router R2 announces routes to `/S1/` content items, however, instead of forwarding the requests to `/S1/`, it simple discards requests making pending requests in previous routers to expire.

Routing information entries can also be threatened by ***passive monitoring***, since a malicious router can announce routes to many content items, but does not keep all of them in its cache. The goal here is to intercept any request for such content and then forward it to the correct location. As the user is not directly affected by this attack, the attacker can monitor requested content in the network without been noticed [111]. This attack could be more dangerous when considering a collaborative network, such as mobile ad hoc networks, in which user devices are naturally placed as routers to forward packets.

Gasti et al. [34] argue that the NDN/CCN architecture is natively resilient against prefix hijacking attacks mainly because routing updates are signed, thus verifiable, preventing malicious users to register fake routes. However, this countermeasure does not apply in face of malicious routers having valid cryptographic keys, accepted and validated by other routers. Furthermore, a network facing such attack could benefit from ***monitoring for anomalous behavior***, similar to solutions applied to request flooding attacks. Also, route hijacking could be subverted by multiple routes toward the victim content, though it would not be effective if attackers collude and attack the publisher in a distributed way [111]. Finally, Lauinger [65] gives some suggestions to prevent the misuse of special bits in request packets, as to limit the use of such bits, like allowing the use of scope field only on local requests, or requiring a digital signature for requests wishing to set the scope field, thus exposing the identification of the attacker.

Table 2 provides a compilation on attacks and countermeasures in ICN routing aspects, together with the vulnerabilities they explore and the targeted entity.

5 SECURITY IN CACHING

The caching mechanism is one of the most prominent features of ICN architectures. It aims to improve content distribution in the network, by placing content copies near users; this alleviates congestion and latency problems, specially for popular content, like video streaming. There is a well-known set of threats against cache systems in traditional networks. However, in ICN architectures these threats are amplified, as they scale globally to the Internet: malicious network entities (publishers and routers) can announce, update, and distribute malicious content that could be very difficult to detect.

Table 2. Overview and Classification of ICN Attacks and Countermeasures on Routers

Attack	Category	Vulnerability	Countermeasures
Fooling rate limit	Resource exhaustion	Limited storage for request state	[3, 7–9, 15, 21, 23, 34, 43, 56, 58, 89, 116]
Flooding			
Pending Interest Table (PIT) attack			
Mobile blockade			
Burst			
Long duration attack			
Jamming			
Forcing expensive computations			
Timeout			
DoS by filling available memory			[65]
Bandwidth depletion	Publisher unavailability		[23, 65]
Single-target DDoS			
DoS against content publisher			
DoS with special *bits*			
Remotely initiated overload			
Piling request due to slow publisher			
FIB pollution	Route depletion	Unauthenticated route announcements	[34, 65, 111]
Route hijacking			
Route-to-death			
Blackhole by prefix hijacking			
Infringing content states			
Route interception			
Routing misuse			

We organize attacks and countermeasures for caching attacks in ICN in three groups: *cache snooping*, *cache pollution*, and *cache poisoning*. Cache snooping attacks comprise malicious users aiming to profile user behavior on the Internet by analyzing content items stored in caches. Cache pollution groups attacks whose goals are to fill caches with unpopular or irrelevant content rather than popular content, making caches ineffective. Finally, cache poisoning differs from previous group by tackling attacks aiming to populate caches with illegal, fake, or forged content. This section lists the attacks that have been explored in literature regarding cache systems for ICN as well as solutions to mitigate them. Since great attention has been given to mitigate caching attacks in ICN, due to the extensive knowledge about caching in traditional networks, there are plentiful solutions.

5.1 Cache Snooping

One of the most interesting cache policies used in ICN is storing content based on content popularity and relevance for nearby users. However, this policy makes the cache a relevant representative of information about users' interest on the Internet, which can be explored to obtain sensitive information about a user or a group of users [83]. Although these attacks can be launched by users in general as it does not require much computational power, content publishers may better benefit from snooping caches to infer users' behavior. Even though it is difficult to pinpoint the behavior of a specific user, content publishers may use snooping to infer preferences for users in a region and use it for advertisement, for example. Also, government agencies may use snooping for

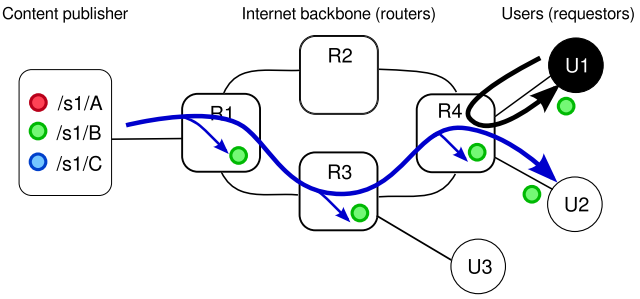


Fig. 9. Cache snooping threat in ICN.

monitoring and blocking specific content. Malicious users closer to victims (e.g., in the same access router) pose an even greater risk for users' privacy, as they share the same cache with fewer users, facilitating snoopers' actions [91]. The snooper is able to ***list cached content, monitor content access, and even copy conversations*** [65], as shown in Figure 9, where user U1 snoops into router R4 for content items stored in cache, retrieving a copy of the content previously requested by user U2.

Strategies for cache snooping include cache probing by monitoring a content name and requesting a specific content name until it is returned by the cache and by requesting random names to the cache, excluding those already obtained in subsequent requests. One particularly worrisome aspect, specific for the NDN/CCN architecture, is the *exclude* field [53], since it could be used to limit a query specifically to the first cache in the path[2]; if a content item is retrieved by the snooper from a specific cache, then a user under that cache has recently retrieved that content [65]. These threats are more worrisome when considering on-path caches, since they are traditionally based on routers and represent users' behavior with more accuracy. Besides considering malicious entities that inspect caches to infer users' behavior, ***privacy invasion attacks*** [73] also consider malicious content publishers that leak users' sensitive information, although this is not a threat exclusive to ICN or under control of ICN architectures.

Another approach to infer whether a given content is in a cache and, consequently, if a nearby user accessed such content recently is through the ***time difference between replies from nearby caches and from the publisher*** [16, 86]. A malicious user can issue probes to the cache in order to measure its Round-Trip Time (RTT). Then, it issues a request for the content it aims to monitor and also measures its RTT. By analyzing both RTTs, the malicious user infers whether the content was retrieved from the cache or from anywhere else [65]. This strategy can also be used to infer if a publisher produced some specific content lately by probing the publisher to discover its RTT and then requesting the monitored content. If the RTT from the second request is lower than the first one, it is safe to deduce that such content was recently made available by the publisher for some user and was stored in the cache of some router. Although this attack is feasible and poses a threat against user privacy and anonymity, it is consensus that correlating a specific user with a content in cache is not trivial and may require additional information [2].

Such attacks pose a greater privacy risk if governments or industries make efforts to spy users and disclose their content access privacy and anonymity by ***monitoring content requests***. This is a fundamental concern for ICN architecture, since the network is aware of content items traveling over it. It can be even worse if routers are able to semantically interpret content names, since

[2]Due to this and other concerns, the *exclude* field is being removed from the current NDN protocol specification (0.3 at this time) [87].

censorship and monitoring by malicious ISPs, governments, and industries would be much easier. In contrast to deep packet inspection tools in traditional networks, in which the snooper is required to be strategically located and powerful enough to inspect packets at line speed, the cache in ICN architectures allows snoopers to retrieve information on a longer time window [16]. However, correlating users and cached copies is not trivial and may require additional information.

As the first countermeasures to tackle cache snooping attacks in ICN paradigm, a set of possible preventive actions were proposed to alleviate the vulnerabilities in ICN for such attack, such as the ***restriction of content name parts in which the user could use the exclude field and the longest prefix matching***, thus setting which part of the name content the user must have prior knowledge to request it [65, 66]. Also, disabling or limiting the use of the exclude field in request packets could prevent malicious users from restricting their search to local caches, where the threat is amplified. Besides prevention, detection actions like ***monitoring events*** such as unusual requests from the same link in a very short time window at a very high hit rate is also an alternative [91].

In face of timing-based cache snooping, the basic strategy is to ***delay replies from caches*** in order to equalize response times, thus avoiding malicious users to infer cached content by timing differences between content retrieved from publishers and caches [16, 65, 85, 86]. Another idea is to count how many times a user requests the same content; a request from the same user is replied directly from the cache and for different users the timing delay is applied, based on the number of hops from the publisher to the user. The use of ***collaborative caches*** can be used to undermine timing attacks, since routers cache content items based on internal state, like available storage, and their position in the forwarding path. Thus, it would be much harder for the malicious user to infer whether the retrieved content was in cache or not.

In order to avoid malicious users from snooping content produced by point-to-point conversations, the main idea is to ***turn content names unpredictable***, e.g., by using ephemeral names, by tunneling requests, or by appending a random component to the content name [2]. This random component needs to be agreed between communicating parties. They choose and securely share a secret seed and use it to create and request content. Thus, content name is only known by the two communicating parties, preventing malicious users to probe the cache. Also, hiding content names from the network may help to avoid malicious users from inferring which content items are relevant for users; however, increased privacy comes in detriment of performance [83].

5.2 Cache Pollution

As populating caches in ICN architectures may depend on users' access pattern, it also raises the possibility of malicious users to populate the cache with uninteresting content, thus disrupting the performance gain brought by caches. Moreover, all users could potentially serve as caches to the network, as in mobile networks in which all nodes are potentially engaged on routing and forwarding tasks, meaning they have unconditional control over their cached content and cache access. A ***cache pollution attack*** [43] is non-intrusive in the sense that malicious users' interactions appear like normal interactions. For example, malicious users could request a large amount of unpopular content in order to disrupt content locality and degrade cache efficiency [22, 65, 120]. On-path caching is more vulnerable to such attack, since routers have limited memory for caching purposes; if irrelevant content items are cached, more requests for valid content will be forwarded to the content publisher. On the other hand, to pollute off-path caches, the attack has to be amplified by colluding a large amount of users to maximize the amount of requests to unpopular content and fake popular interest. Figure 10 illustrates two users, U1 and U3, colluding to pollute caches between them. In this scenario, U3 requests unpopular content from U1, leaving a cached copy of this uninteresting content on caches of R4 and R3, which may displace a legitimate popular content.

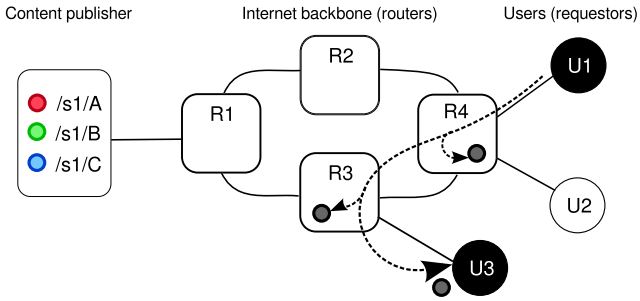


Fig. 10. Cache pollution threat in ICN.

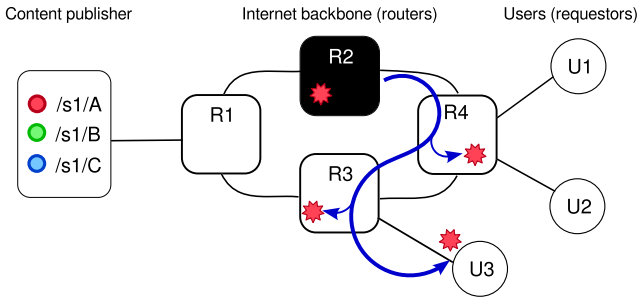


Fig. 11. Cache poisoning threat in ICN.

Monitoring content on caches is the most efficient solution for avoiding cache pollution [57, 121]. For example, the same monitoring mechanisms used for detecting and monitoring other attacks may be used to prevent cache pollution, since they already foresee cache monitoring metrics [22, 43, 120]. Specifically, the monitoring of incoming content requests to the same prefix in a short period is especially useful on monitoring cache pollution attacks [121].

5.3 Cache Poisoning

Cache poisoning attacks pose a serious threat to ICN architectures, since their main advantage is easy content production and distribution. A malicious user could populate its cache and satisfy legitimate requests with forged content, which will subsequently be cached along the forwarding path. Moreover, if the content name for a popular content could be anticipated, a malicious user could potentially produce fake content beforehand and collude with others to request this content, in order to spread it along the caches. Then routers with content in cache will deliver the fake content for legitimate requests when the original content becomes available and legitimate users start to request it [30, 34, 38, 73]. However, polluting caches is not trivial, may require additional preparation from attackers, such as subverting the routing system [68], and may require greater computational power, since they should spread fake content toward caches as much as possible. For example, Figure 11 exemplifies the case in which router *R2* anticipates the name of a popular content; it produces and distributes the content for inadvertent users while populating caches with the forged content.

Concerning fake content distribution, ICN relies on policies for cache content eviction or explicit exclusion of undesired content by users, which is usually not sufficient to avoid cache poisoning [38]. Having routers to ***verify all content items signatures*** before caching them is the natural countermeasure, as it prevents the router to cache fake content. However, this may severely

impact routing performance, as the router would need to check signatures for all incoming content, and a trust mechanism should also be deployed (possibly application-specific). Alternatively, routers may verify signatures only when they are requested to serve content from their caches, preventing fake content to spread into other routers and the users [51]. Also, ***using information about content excluded from users requests*** would lead to the same problems: the router would need to authenticate the user to avoid that malicious users deliberately exclude valid content [39, 65]. Another approach to avoid cache poisoning consists in deploying ***blacklists*** with illegal content names; routers can then deny their delivery and periodically re-validate cache entries with the content publishers [65]. However, more robust countermeasures seem necessary to effectively avoid fake content on caches.

The basic approach of countermeasures to cache poisoning attacks is to ***check content provenance***; for example, routers may accept to cache content items only if the incoming interface matches the outgoing interface in which the request was issued [51]. Assuming that users have means of knowing the content name beforehand, one solution is to include the hash of the static content as an alternative component in a content request [34]. Thereby, upon the reception of such content, the user could be sure the content is the right answer for her request. For dynamic content, the idea is to include the hash of the content publisher's public key into the request; each intermediate router should check whether the returned content references the same public key. However, in this scenario, malicious users could collude with a malicious content publisher that signs every content with a distinct cryptographic key, forcing the router to ask for every key and consequently delaying content signature checking. Also, the malicious publisher could delay sending the key to the router and choose keys and signatures that require more computational resources to be verified. This scenario leads the router to deny services to legitimate users or even to stop verifying signatures, inducing a momentary security breach.

Routers could stop checking content signatures if the load become too high or, in case of facing a suspicious content publisher, to verify content signatures only when the router has enough free processing power [35, 37, 39]; however, it may lead to an unnecessary performance loss. Alternatively, they propose signature checks only if the content is stored long enough in cache [65]. Since in NDN/CCN data packets are signed, DiBenedetto and Papadopoulos [27] propose that end hosts report their routers about the bad data packets they detect. Each router then verifies the packet signature and, if needed, removes it from its cache, forwards the report upstream, and adjusts its forwarding strategies to avoid that content's source.

Table 3 lists the last part of security attacks and countermeasures for ICN, grouping attacks in ICN caching mechanisms and their corresponding category, target, vulnerability, as well as countermeasures to mitigate them.

6 CHALLENGES AND FUTURE PERSPECTIVES

Effectively securing computing systems in general is not a trivial task [100]. The Internet poses an extra challenge, as its pervasiveness and ubiquitous presence give users the sense of anonymity. As more applications are available on the Internet, more users expose information regarding many facets of their lives, such as credit card numbers, bank accounts, addresses, daily routines, aspects about private life, and so on. With the advent of the Internet of Things [81], it is expected that everything will eventually be connected to the Internet, from appliances to health devices. Thus, the security aspect becomes even more worrying, as sensitive content (e.g., concerning user's health) requires guarantee that only authorized users can access and modify it.

As stated by AbdAllah et al. [1], ICN security solutions should ensure content integrity and authenticity, certify content provenance, and protect user privacy. Considering also availability and scalability issues, this scope statement is not easy to be fulfilled. While ICN itself is natively

Table 3. Overview and Classification of ICN Attacks and Countermeasures on Caching

Attack	Category	Vulnerability	Countermeasures
Cache snooping	Cache snooping	Cache replies to any user	[65, 91]
Timing attacks			[16, 65, 86]
Request monitoring			[2, 16, 66, 67, 83]
Object discovery			
Data flow cloning			
Monitoring and censorship			[16]
Cache pollution	Cache pollution	Cache acceptance policy	[22, 43, 65, 120, 121]
Content Store (CS) DoS			
Locality disruption			
False locality			
Cache poisoning	Cache poisoning		[27, 34, 35, 37–39, 51, 65, 97]
False content injection			
Content falsification			
Unwanted content on caches			
Cache misuse			
Spam			[30]

prone, to a certain extent, to some well-known attacks (such as DoS), other important security aspects should be addressed before it can be deployed in real world. In the following, we discuss some ideas we think are important to promote for the evolution of security foundations in ICN:

Old threats in a new architecture: ICN security area would benefit from a deep inspection concerning the feasibility and impact of attacks already known in IP-based networks on ICN architectures, as web page hijacking, traffic interception, and spam, for example. It would be helpful to thoroughly explore these attacks to discover new vulnerabilities and also defenses against them, possibly provided by the ICN paradigm itself.

Security vulnerabilities in mobile environments: Mobile environments pose an extra challenge for security in ICN, since all devices are supposed to serve as content caches, thus raising the potential for cache snooping, monitoring, and censorship. Works addressing mobile ICN security vulnerabilities and threats are still scarce. There should be more investigation about the impact of the ICN paradigm in a mobile environment, as ICN may be vulnerable to current mobile threats or even introduce new forms of vulnerabilities in mobile networks.

User privacy versus malicious activity detection: Previous cases concerning unauthorized monitoring and spying on users' Internet traffic have raised the level of worry about user privacy and anonymity. In order to be fully considered as an alternative to the current Internet, the ICN architecture must comply with users rights and international rules concerning user privacy. It is not easy to reason about implications of such mechanisms, and it is not clear how to simultaneously enable both privacy and malicious activity detection. Moreover, ensuring that governments collect users' traffic patterns only for security reasons and not for censorship, or preventing misuse of such information to coerce or threat individuals, goes beyond what technology can offer: It requires users behaving properly and citizens vigilance over government activities.

Attack impact assessment: A deep understanding of the potential of attacks in ICN is a considerable challenge. We have listed several research works suggesting potential security

problems in ICN; however, only few of them effectively show and analyze the impact of such attacks, in simulations or in a real ICN deployments. Such analysis and discussions are important to really understand which aspects need further study and then correctly orientate the research toward more secure ICN architectures.

Better evaluation of solutions: To date, many solutions proposed in the surveyed literature are not deeply evaluated. More realistic and detailed evaluations are crucial to better understand the behavior of the proposed solutions, including effectiveness, overhead, side effects, and trade-offs [6]. Many testbeds have been developed for ICN architectures [46, 80, 98, 105], and they should be more extensively used to provide better knowledge about ICN security solutions. In particular, solutions that hamper content caching or limit request rates may negatively affect the user experience and should be better investigated [107].

Scalability issues: Many of the security solutions described here directly affect scalability, the *raison d'être* and central feature in ICN architectures. For instance, some solutions use centralized entities for access control enforcement [31, 32, 36, 102] or for key authenticity verification [96]. Scalability should be a foreground property in the assessment of security solutions for attacks in the ICN context.

Fault tolerance: Some of the proposed solutions are based in third-party entities that may constitute a single point of failure [32]. Such entities, besides impacting the system scalability, may constitute interesting targets for DoS attacks, impairing the whole network. Some controlled redundancy is expected from solutions to leverage fault tolerance and scalability.

Trust management: The binding between a key and its owner should be ensured by a trust mechanism. ICN does not impose a specific trust management scheme, leaving to applications to define their own mechanisms and sources of trust. Approaches used in the current Internet, like hierarchical Public Key Infrastructures (PKI) can also be adopted in ICN, but more scalable approaches, like SPKI/SDSI [19], fit better to the decentralized nature of ICN, as stated by Zhang et al. [128].

Key revocation: Besides checking the bound between a key and its owner, clients also need to verify whether a given key is yet valid (i.e., it was not revoked). Since a key is a data content, it can be stored in and retrieved from network caches and thus may be outdated if its issuer revoked it. Some approaches were proposed to check the status of keys in ICN [82, 96, 123], but more generic solutions, possibly integrated with trust management schemes, would be welcome.

The ICN paradigm gives researchers the opportunity to embed security into the core of the Internet and to rethink what worked and what has been unsuccessful in the traditional Internet architecture. Nonetheless, securing the network will always be a continuous effort, as new threats appear, new attacks are explored, and new security measures are provided. Starting with robust and consolidated security foundations embedded into the architecture, researchers will be able to provide better and quicker countermeasures, turning the Internet into a safer place.

7 CONCLUDING REMARKS

The ICN paradigm is a promising future Internet architecture that comes to provide an Internet more suitable to content distribution. However, from the security perspective, ICN features great challenges. In this article, we organized, classified, discussed, and explored the literature about security vulnerabilities, attacks, and countermeasures in ICN, regarding three key ICN aspects: content, routers, and caching. We observed that while severe vulnerabilities on content and

content names, such as integrity and privacy, have been extensively addressed, threats against routers and caching still needs attention, especially when they allow fake or corrupted content to reach users. Thus, further research is needed on security solutions and metrics, to leverage ICN as a complete, robust architecture, able to meet current and future demands for performance, mobility, and security. We expect that this knowledge systematization can provide the reader to insights regarding ICN attacks and vulnerabilities, helping into building future works on discovering points of vulnerabilities and new solutions and countermeasures to mitigate them.

ACKNOWLEDGMENTS

Dr. Elisa Mannes would also like to thank Dr. Cinara Menegazzo and Dr. Rebeca Schroeder for their immeasurable support and motivation to begin this work.

REFERENCES

[1] Eslam AbdAllah, Hossam Hassanein, and Mohammad Zulkernine. 2015. A survey of security attacks in information-centric networking. *IEEE Communications Surveys Tutorials* 17, 3 (2015), 1441–1454.

[2] Gergely Acs, Mauro Conti, Paolo Gasti, Cesar Ghali, and Gene Tsudik. 2013. Cache privacy in named-data networking. In *International Conference on Distributed Computing Systems (ICDCS'13)*. IEEE, 41–51.

[3] Alexander Afanasyev, Priya Mahadevan, Ilya Moiseenko, Ersin Uzun, and Lixia Zhang. 2013. Interest flooding attack and countermeasures in named data networking. In *International Conference on Networking (Networking'13)*. IFIP, 1–9.

[4] Alexander Afanasyev, Cheng Yi, Lan Wang, Beichuan Zhang, and Lixia Zhang. 2015. SNAMP: Secure namespace mapping to scale NDN forwarding. In *IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, 281–286.

[5] Bengt Ahlgren, Christian Dannewitz, Claudio Imbrenda, Dirk Kutscher, and Börje Ohlman. 2012. A survey of information-centric networking. *IEEE Communications Magazine* 50, 7 (2012), 26–36.

[6] Samir Al-Sheikh, Matthias Wählisch, and Thomas C. Schmidt. 2015. Revisiting countermeasures against NDN interest flooding. In *ACM Conference on Information-Centric Networking (ICN'15)*. ACM, 195–196.

[7] Bander Alzahrani, Vassilios Vassilakis, and Martin Reed. 2013. Key management in information centric networking. *International Journal of Computer Networks and Communications (IJCNC)* 5 (2013), 153–166.

[8] Bander Alzahrani, Vassilios Vassilakis, and Martin Reed. 2013. Mitigating brute-force attacks on Bloom-filter based forwarding. In *Conference on Future Internet Communications (CFIC'13)*. IEEE, 1–7.

[9] Bander Alzahrani, Vassilios Vassilakis, and Martin Reed. 2013. Securing the forwarding plane in information centric networks. In *Computer Science and Electronic Engineering Conference (CEEC'13)*. IEEE, 174–178.

[10] Somaya Arianfar, Teemu Koponen, Barath Raghavan, and Scott Shenker. 2011. On preserving privacy in content-oriented networks. In *ACM SIGCOMM Workshop on ICN (ICN'11)*. ACM, 19–24.

[11] Giuseppe Ateniese, Kevin Fu, Matthew Green, and Susan Hohenberger. 2006. Improved proxy re-encryption schemes with applications to secure distributed storage. *ACM Transactions on Information System Security* 9, 1 (2006), 1–30.

[12] Mohammed Bari, Shihabur Chowdhury, Reaz Ahmed, Raouf Boutaba, and Bertrand Mathieu. 2012. A survey of naming and routing in information-centric networks. *IEEE Communications Magazine* 50, 12 (2012), 44–53.

[13] Mark Baugher, Bruce Davie, Ashok Narayanan, and David Oran. 2012. Self-verifying names for read-only named data. In *Conference on Computer Communications Workshops (INFOCOM)*. IEEE, 274–279.

[14] Flavio Bonomi, Michael Mitzenmacher, Rina Panigrahy, Sushil Singh, and George Varghese. 2006. An improved construction for counting bloom filters. In *Annual European Symposium on Algorithms (ESA'06)*. Springer, Zurich, Switzerland, 684–695.

[15] Jeff Burke, Paolo Gasti, Naveen Nathan, and Gene Tsudik. 2014. Secure sensing over named data networking. In *International Symposium on Network Computing and Applications (NCA'14)*. IEEE, 175–180.

[16] Abdelberi Chaabane, Mohamed Ali De Cristofaro, Emiliano andiaafar, and Ersin Uzun. 2013. Privacy in content-oriented networking: Threats and Countermeasures. *SIGCOMM Computer Communications Review* 43, 3 (2013), 25–33.

[17] Tao Chen, Kai Lei, and Kuai Xu. 2014. An encryption and probability based access control model for named data networking. In *IEEE International Performance Computing and Communication Conference (IPCCC'14)*. IEEE, 1–8.

[18] Seungoh Choi, Kwangsoo Kim, Seongmin Kim, and Byeong hee Roh. 2013. Threat of DoS by interest flooding attack in content-centric networking. In *International Conference on Information Networking (ICOIN'13)*. IEEE, 315–319.

[19] Dwaine Clarke, Jean-Emile Elien, Carl Ellison, Matt Fredette, Alexander Morcos, and Ronald Rivest. 2001. Certificate chain discovery in SPKI/SDSI. *Journal of Computer Security* 9, 4 (2001), 285–322.

[20] Alberto Compagno, Mauro Conti, Paolo Gasti, Luigi Vicenzo Mancini, and Gene Tsudik. 2015. Violating consumer anonymity: Geo-locating nodes in named data networking. In *International Conference on Applied Cryptography and Network Security (ACNS'15)*. Springer.

[21] Alberto Compagno, Mauro Conti, Paolo Gasti, and Gene Tsudik. 2013. Poseidon: Mitigating interest flooding DDoS attacks in named data networking. In *Conference on Local Computer Networks (LCN'13)*. IEEE, 1–9.

[22] Mauro Conti, Paolo Gasti, and Marco Teoli. 2013. A lightweight mechanism for detection of cache pollution attacks in named data networking. *Computer Networks* 57, 16 (2013), 3178–3191.

[23] Huichen Dai, Yi Wang, Jindou Fan, and Bin Liu. 2013. Mitigate DDoS attacks in NDN by interest traceback. In *IEEE International Workshop on Emerging Design Choices in Name-Oriented Networking (NOMEN'13)*. IEEE, 381–386.

[24] Christian Dannewitz, Jovan Golić, Börje Ohlman, and Bengt Ahlgren. 2010. Secure naming for a network of information. In *IEEE Conference on Computer Communications Workshops (INFOCOM'10)*. IEEE, 1–6.

[25] Christian Dannewitz, Dirk Kutscher, Börje Ohlman, Stephen Farrell, Bengt Ahlgren, and Holger Karl. 2013. Network of information (NetInf) - An information-centric networking architecture. *Comp. Communications* 36, 7 (2013), 721–735.

[26] Steve DiBenedetto, Paolo Gasti, Gene Tsudik, and Ersin Uzun. 2012. ANDaNA: Anonymous named data networking application. In *Network and Distributed System Security Symposium (NDSS'12)*. Internet Society.

[27] Stephanie DiBenedetto and Christos Papadopoulos. 2016. Mitigating poisoned content with forwarding strategy. In *Conference on Computer Communications Workshops*. IEEE, 164–169. DOI: https://doi.org/10.1109/INFCOMW.2016.7562065

[28] Onyekachi O. Elechi, Joseph S. Igwe, and Elias C. Eze. 2014. Denial of service in internet protocol network and information centric network: An impediment to network quality of service. *Journal of Information Engineering and Applications* 4 (2014), 14–24.

[29] Stephen Farrell, Dirk Kutscher, Christian Dannewitz, Börje Ohlman, Ari Keränen, and Phillip Hallam-Baker. 2013. Naming Things with Hashes. RFC 6920.

[30] Nikos Fotiou, Giannis F. Marias, and George C. Polyzos. 2010. Towards a secure rendezvous network for future publish/subscribe architectures. In *Future Internet Symposium (FIS'10)*. Springer, Berlin, Germany, 49–56.

[31] Nikos Fotiou, Giannis F. Marias, and George C. Polyzos. 2012. Access control enforcement delegation for information-centric networking architectures. In *ACM SIGCOMM Workshop on ICN (ICN'12)*. ACM, 85–90.

[32] Nikos Fotiou, Yannis Thomas, Vasilios A. Siris, and George C. Polyzos. 2014. Security requirements and solutions for integrated satellite-terrestrial information-centric networks. In *Advanced Satellite Multimedia Systems Conference, Signal Processing for Space Communications Workshop (ASMS/SPSC)*. IEEE, 1–8.

[33] Nikos Fotiou, Dirk Trossen, Giannis Marias, Alexandros Kostopoulos, and George Polyzos. 2013. Enhancing information lookup privacy through homomorphic encryption. *Journal of Security and Communication Networks* 7 (2013), 2804–2814.

[34] Paolo Gasti, Gene Tsudik, Ersin Uzun, and Lixia Zhang. 2012. DoS and DDoS in named-data networking. In *International Conference on Computer Communications and Networks (ICCCN'13)*. IEEE, 1–7.

[35] Cesar Ghali, Ashok Narayanan, David Oran, and Gene Tsudik. 2014. Secure Fragmentation for Content-Centric Networks. http://arxiv.org/abs/1405.2861.

[36] Cesar Ghali, Marc A. Schlosberg, Gene Tsudik, and Christopher A. Wood. 2015. Interest-based access control for content centric networks. In *ACM Conference on Information-Centric Networking (ICN'15)*. ACM, 147–158.

[37] Cesar Ghali, Gene Tsudik, and Ersin Uzun. 2014. Elements of trust in named-data and content-centric networking. *ACM SIGCOMM Computer Communication Review* 44, 5 (2014), 1–10.

[38] Cesar Ghali, Gene Tsudik, and Ersin Uzun. 2014. Needle in a haystack: Mitigating content poisoning in named-data networking. In *Workshop on Security of Emerging Networking Technologies (SENT'14)*. Internet Society, 1–10.

[39] Cesar Ghali, Gene Tsudik, and Ersin Uzun. 2014. Network-layer trust in named-data networking. *SIGCOMM Computer Communications Review* 44, 5 (2014), 12–19.

[40] Cesar Ghali, Gene Tsudik, Ersin Uzun, and Christopher A. Wood. 2015. Living in a PIT-less world: A case against stateful forwarding in content-centric networking. *ArXiv* 1512, 07755 (2015), 1–10. http://arxiv.org/abs/1512.07755.

[41] Cesar Ghali, Gene Tsudik, and Christopher Wood. 2016. Network names in content-centric networking. In *3rd ACM Conference on Information-Centric Networking (ACM-ICN'16)*. ACM, 132–141. DOI: https://doi.org/10.1145/2984356.2984373

[42] Cesar Ghali, Gene Tsudik, and Christopher A. Wood. 2017. When encryption is not enough: Privacy attacks in content-centric networking. In *ACM Conference on Information-Centric Networking (ICN'17)*. ACM, 1–10.

[43] David Goergen, Thibault Cholez, Jérôme François, and Thomas Engel. 2012. Security monitoring for content-centric networking. In *International Workshop on Data Privacy Management (DPM'12)*. Springer, 274–286.

[44] David Goergen, Thibault Cholez, Jérôme François, and Thomas Engel. 2013. A semantic firewall for content-centric networking. In *International Symposium on Integrated Network Management (IM'13)*. IFIP/IEEE.

[45] Aaron D. Goldman, A. Selcuk Uluagac, and John A. Copeland. 2014. Cryptographically-curated file system (CCFS): Secure, inter-operable, and easily implementable information-centric networking. In *Conference on Local Computer Networks (LCN'14)*. IEEE, 142–149.

[46] Pedro Henrique Guimaraes, Lyno Henrique Ferraz, João Vitor Torres, Diogo Mattos, Andres Murillo, Martin Andreoni, Igor Alvarenga, Claudia Rodrigues, and Otto Carlos Duarte. 2013. Experimenting content-centric networks in the future internet testbed environment. In *Workshop on Cloud Convergence: Challenges for Future Infrastructures and Services (WCC'13)*. IEEE, 1403–1407.

[47] Balkis Hamdane and Sihem Guemara El Fatmi. 2015. A credential and encryption based access control solution for named data networking. In *International Symposium on Integrated Network Management (IM'15)*. IEEE, 1234–1237.

[48] Balkis Hamdane, Sihem Guemara El Fatmi, and Ahmed Serhrouchni. 2014. A novel name-based security mechanism for information-centric networking. In *Wireless Communications and Networking Conference (WCNC'14)*. IEEE, 1–5.

[49] Balkis Hamdane, Mounira Msahli, Ahmed Serhrouchni, and Sihem Guemara El Fatmi. 2013. Data-based access control in named data networking. In *IEEE International Conference on Collaborative Computing*. IEEE, 531–536.

[50] Balkis Hamdane, Ahmed Serhrouchni, Ahmad Fadlallah, and Sihem El Fatmi. 2012. Named-data security scheme for named data networking. In *International Conference on the Network of the Future (NOF'12)*. IEEE, 1–6.

[51] Do Hyung Kim, SunWook Nam, Jun Bi, and Ikjun Yeom. 2015. Efficient content verification in named data networking. In *ACM Conference on Information-centric Networking (ICN'15)*. ACM, 109–116.

[52] Mihaela Ion, Jianqing Zhang, and Eve Schooler. 2013. Toward content-centric privacy in ICN: Attribute-based encryption and routing. In *ACM SIGCOMM Workshop on ICN (ICN'13)*. ACM, 39–40.

[53] Van Jacobson, Diana K. Smetters, James D. Thornton, Michael Plass, Nick Briggs, and Rebecca Braynard. 2012. Networking named content. *Communications of the ACM* 55, 1 (2012), 117–124.

[54] Van Jacobson, Diana K. Smetters, James D. Thornton, Michael F. Plass, Nicholas H. Briggs, and Rebecca L. Braynard. 2009. Networking named content. In *International Conference on Emerging Networking Experiments and Technologies (CoNEXT'09)*. ACM, 1–12.

[55] Jongmin Jeong, Ted "Taekyoung" Kwon, and Yanghee Choi. 2010. Host-oblivious security for content-based networks. In *International Conference on Future Internet Technologies (CFI'10)*. ACM, 35–40.

[56] Amin Karami. 2013. Data clustering for anomaly detection in content-centric networks. *International Journal of Computer Applications* 81, 7 (2013), 1–8.

[57] Amin Karami and Manel Guerrer-Zapata. 2015. An ANFIS-based cache replacement method for mitigating cache pollution attacks in named data networking. *Computer Networks* 80 (2015), 51–65.

[58] Amin Karami and Manel Guerrero-Zapata. 2015. A fuzzy anomaly detection system based on hybrid PSO-kmeans algorithm in content-centric networks. *Neurocomputing* 149 (2015), 1253–1269.

[59] Amin Karami and Manel Guerrero-Zapata. 2015. A hybrid multiobjective RBF-PSO method for mitigating DoS attacks in named data networking. *Neurocomputing* 151, Part 3 (2015), 1262–1282.

[60] Fawad Khan, Sarmad Ullah Khan, Muhammad Roman, and Usman Abbasi. 2014. Location identity based content security scheme for content centric networking. In *International Conference on Security of Information and Networks (SIN'14)*. ACM, 1–6.

[61] Sarmad Ullah Khan, Thibault Cholez, Thomas Engel, and Luciano Lavagno. 2013. A key management scheme for content centric networking. In *International Symposium on Integrated Network Management (IM'13)*. IFIP/IEEE, 828–831.

[62] Teemu Koponen, Mohit Chawla, Byung-Gon Chun, Andrey Ermolinskiy, Kye Hyun Kim, Scott Shenker, and Ion Stoica. 2007. A data-0riented (and beyond) network architecture. *SIGCOMM Computer Communications Review* 37, 4 (2007), 181–192.

[63] Jun Kurihara, C. Wood, and Ersin Uzun. 2015. An encryption-based access control framework for content-centric networking. In *IFIP Networking*. IEEE, 1–9.

[64] Dirk Kutscher, Suyong Eum, Kostas Pentikousis, Ioannis Psaras, Daniel Corujo, Damien Saucez, Thomas Schmidt, and Matthias Wählisch. 2016. Information-Centric Networking (ICN) Research Challenges. RFC 7927. DOI: https://doi.org/10.17487/RFC7927

[65] Tobias Lauinger. 2010. *Security and Scalability of Content-Centric Networking*. Master's thesis. Eurecom, Sophia-Antipolis, France and Technische Universität Darmstadt, Germany.

[66] Tobias Lauinger, Nikolaos Laoutaris, Pablo Rodriguez, Thorsten Strufe, Ernst Biersack, and Engin Kirda. 2012. Privacy risks in named data networking: What is the cost of performance? *SIGCOMM Computer Communications Review* 42, 5 (2012), 54–57.

[67] Tobias Lauinger, Thorsten Strufe, Nikolaos Laoutaris, Ernst Biersack, Pablo Rodriguez, and Engin Kirda. 2011. *Privacy Implications of Ubiquitous Caching in Named Data Networking Architectures*. Technical Report. Technische Universität Darmstadt. TR-iSecLab-0812-001.

[68] Vince Lehman, AKM Mahmudul Hoque, Yingdi Yu, Lan Wang, Beichuan Zhang, and Lixia Zhang. 2016. *A Secure Link State Routing Protocol for NDN*. Technical Report NDN-0037. NDN. Retrieved from http://named-data.net/techreports.html.

[69] Bing Li, Ashwin Prabhu Verleker, Dijiang Huang, Zhijie Wang, and Yan Zhu. 2014. Attribute-based access control for ICN naming scheme. In *IEEE Conference on Communications and Network Security (CNS'14)*. IEEE, 391–399.

[70] Bing Li, Zhijie Wang, Dijiang Huang, and Yan Zhu. 2014. *Toward Privacy-preserving Content Access Control for Information Centric Networking*. Technical Report. Arizona State University.

[71] Qi Li, Ravi Sandhu, Xinwen Zhang, and Mingwei Xu. 2015. Mandatory content access control for privacy protection in information centric networks. *IEEE Transactions on Dependable and Secure Computing* PP, 99 (2015), 1–13.

[72] Qi Li, Xinwen Zhang, Qingji Zheng, Ravi Sandhu, and Xiaoming Fu. 2014. LIVE: Lightweight integrity verification and content access control for named data networking. *IEEE Transactions on Information Forensics and Security* 10, 2 (2014), 308–320.

[73] Jonathan Loo and Mahdi Aiash. 2015. Challenges and solutions for secure information-centric networks: A case study of the netinf architecture. *Journal of Network and Computer Applications* 50 (2015), 64–72.

[74] You Lu, Zhiyang Wang, Yu-Ting Yu, Ruolin Fan, and Mario Gerla. 2013. Social network based security scheme in mobile information-centric network. In *Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net'13)*. IEEE, 1–7.

[75] Priya Mahadevan, Ersin Uzun, Spencer Sevilla, and J. J. Garcia-Luna-Aceves. 2014. CCN-KRS: A key resolution service for CCN. In *ACM Conference on Information-centric Networking (ICN'14)*. ACM, 97–106.

[76] Michele Mangili, Fabio Martignon, and Stefano Paraboschi. 2015. A cache-aware mechanism to enforce confidentiality, trackability and access policy evolution in content-centric networks. *Computer Networks* 76 (2015), 126–145.

[77] Elisa Mannes, Carlos Maziero, Luiz Carlos Lassance, and Fàbio Borges. 2015. Optimized access control over encrypted content in information-centric networks. In *IEEE Symposium on Computers and Communications (ISCC'15)*. IEEE, 924–929.

[78] Elisa Mannes, Carlos Maziero, Luiz Carlos Lassance, and Fabio Borges. 2016. Assessing the impact of cryptographic access control solutions on multimedia delivery in information-centric networks. In *Network Operations and Management Symposium (NOMS'16)*. IEEE, 427–435.

[79] Emmanuel A. Massawe, Suguo Du, and Haojin Zhu. 2013. A scalable and privacy-preserving named data networking architecture based on bloom filters. In *International Conference on Distributed Computing Systems Workshops (ICDCSW'13)*. IEEE, 22–26.

[80] Spyridon Mastorakis, Alexander Afanasyev, Ilya Moiseenko, and Lixia Zhang. 2015. *ndnSIM 2.0: A New Version of the NDN Simulator for NS-3*. Technical Report NDN-0028. NDN.

[81] Friedemann Mattern and Christian Floerkemeier. 2010. From the internet of computers to the internet of things. In *From Active Data Management to Event-based Systems and More*. Springer, Heildelberg, Germany, 242–259.

[82] Giulia Mauri and Giacomo Verticale. 2013. Distributing key revocation status in named data networking. In *Advances in Communication Networking*. Lecture Notes in Computer Science, Vol. 8115. Springer, Chemnitz, Germany, 310–313.

[83] Giulia Mauri and Giacomo Verticale. 2014. On the tradeoff between performance and user privacy in information centric networking. In *Conference on New Technologies, Mobility and Security (NTMS'14)*. IEEE, 1–5.

[84] Satyajayant Misra, Reza Tourani, and Nahid Ebrahimi Majd. 2013. Secure content delivery in information-centric networks: Design, implementation, and analyses. In *ACM SIGCOMM Workshop on ICN (ICN'13)*. ACM, 73–78.

[85] Aziz Mohaisen, Hesham Mekky, Xinwen Zhang, Haiyong Xie, and Yongdae Kim. 2014. Timing attacks on access privacy in information centric networks and countermeasures. *IEEE Transactions on Dependable and Secure Computing* 12, 6 (2014), 675–687.

[86] Abedelaziz Mohaisen, Xinwen Zhang, Max Schuchard, Haiyong Xie, and Yongdae Kim. 2012. Protecting access privacy of cached contents in information centric networks. In *ACM Conference on Computer and Communications Security (CCS'12)*. ACM, 1001–1003.

[87] NDN Project. 2017. *Named Data Networking Project Specifications, v 0.3*. Technical Report. Named Data Networking Project. Retrieved from https://named-data.net/project/specifications.

[88] Tan Nguyen, Remi Cogranne, and Guillaume Doyen. 2015. An optimal statistical test for a robust detection of interest flooding attacks in CCN. In *IEEE/IFIP International Symposium on Integrated Network Management*. IEEE, 1–9.

[89] Tan Nguyen, Remi Cogranne, Guillaume Doyen, and Florent Retraint. 2015. Detection of interest flooding attacks in named data networking using hypothesis testing. In *IEEE Workshop on Information Forensics and Security (WIFS'15)*. IEEE, 1–6.

[90] Boubakr Nour, Kashif Sharif, Fan Li, Hassine Moungla, and Yang Liu. 2017. M2HAV: A standardized ICN naming scheme for wireless devices in internet of things. In *12th International Conference on Wireless Algorithms, Systems, and Applications (WASA'17)*. Springer, 289–301.

[91] Nonhlanhla Ntuli and Sunyoung Han. 2012. Detecting router cache snooping in named data networking. In *International Conference on ICT Convergence (ICTC'12)*. IEEE, 714–718.
[92] John P. Papanis, Stavros I. Papapanagiotou, Aziz S. Mousas, Georgios V. Lioudakis, Dimitra I. Kaklamani, and Iakovos S. Venieris. 2013. On the use of attribute-based encryption for multimedia content pprotection over information-centric networks. *Transactions on Emerging Telecommunications Technologies* 25, 4 (2013), 422–435.
[93] Diego Perino and Matteo Varvello. 2011. A reality check for content centric networking. In *ACM SIGCOMM Workshop on ICN (ICN'11)*. ACM, 44–49.
[94] Daniel Posch, Hermann Hellwagner, and Peter Schartner. 2013. On-demand video streaming based on dynamic adaptive encrypted content chunks. In *IEEE International Conference on Network Protocols (ICNP'13)*. IEEE, 1–6.
[95] Eric Renault, Abid Ahmad, and Mohamed Abid. 2009. Toward a security model for the future network of information. In *International Conference on Ubiquitous Information Technologies Applications (ICUT'09)*. IEEE, 1–6.
[96] Daniel Rezende, Carlos Maziero, and Elisa Mannes. 2018. A distributed online certificate status protocol for named data networks. In *ACM Symposium on Applied Computing (SAC'18)*. ACM, 2102–2108.
[97] Igor Ribeiro, Antonio Rocha, Celio Albuquerque, and Flavio Guimarães. 2014. On the possibility of mitigating content pollution in content-centric networking. In *Conference on Local Computer Networks (LCN'14)*. IEEE, 498–501.
[98] Lorenzo Saino, Ioannis Psaras, and George Pavlou. 2014. Icarus: A caching simulator for information centric networking (ICN). In *International Conference on Simulation Tools and Techniques (SIMUTOOLS)*. EAI, 1–10.
[99] Sandvine. 2014. Sandvine global Internet Phenomena Report: 1H 2014. Retrieved from https://www.sandvine.com/downloads/general/global-internet-phenomena/2014/1h-2014-global-internet-phenomena-report.pdf.
[100] Bruce Schneier. 2012. *Liars and Outliers: Enabling the Trust that Society Needs to Thrive* (1st ed.). Wiley, New York NY.
[101] Seog Chung Seo, Taehong Kim, and MyeongWuk Jang. 2014. A privacy-preserving approach in content centric networks. In *Consumer Communications and Networking Conference (CCNC'14)*. IEEE, 866–871.
[102] Sapna Singh, Archana Puri, Shiksha Smreti Singh, Anurika Vaish, and S. Venkatesan. 2012. A trust based approach for secure access control in information centric network. *International Journal of Information and Network Security (IJINS)* 1, 2 (2012), 97–104.
[103] Diana Smetters and Van Jacobson. 2009. *Securing Network Content*. Technical Report. PARC TR-2009-1.
[104] Xiaobin Tan, Zifei Zhou, C. Zou, Yukun Niu, and Xin Chen. 2014. Copyright protection in named data networking. In *6th International Conference on Wireless Communications and Signal Processing (WCSP'14)*. IEEE, 1–6.
[105] Michele Tortelli, Dario Rossi, Gennaro Boggia, and Luigi Alfredo Grieco. 2014. Cross-comparison of ICN software tools. In *ACM Conference on Information-centric Networking (ICN'14)*. ACM, 197–198.
[106] Reza Tourani, Satyajayant Misra, Joerg Kliewer, Scott Ortegel, and Travis Mick. 2015. Catch me if you can: A practical framework to evade censorship in information-centric networks. In *ACM Conference on Information-Centric Networking (ICN'15)*. ACM, 167–176.
[107] Reza Tourani, Satyajayant Misra, Travis Mick, and Gaurav Panwar. 2018. Security, privacy, and access control in information-centric networking: A survey. *IEEE Communications Surveys Tutorials* 20, 1 (2018), 566–600.
[108] Gene Tsudik, Ersin Uzun, and Christopher A. Wood. 2014. AC3N: An API and service for anonymous communication in content-centric networking. In *Consumer Communications and Networking Conference (CCNC'14)*. IEEE, 858–865.
[109] Bárbara Vieira and Erik Poll. 2013. A security protocol for information-centric networking in smart grids. In *Smart Energy Grid Security Workshop (SEGS'13)*. ACM, 1–10.
[110] Matteo Virgilio, Guido Marchetto, and Riccardo Sisto. 2013. PIT overload analysis in content centric networks. In *ACM SIGCOMM Workshop on ICN (ICN'13)*. ACM, 67–72.
[111] Matthias Wählisch, Thomas C. Schmidt, and Markus Vahlenkamp. 2013. Backscatter from the data plane – Threats to stability and security in information-centric network infrastructure. *Computer Networks* 57, 16 (2013), 3192–3206.
[112] Matthias Wählisch, Thomas C. Schmidt, and Markus Vahlenkamp. 2013. Lessons from the past: Why data-driven states harm future information-centric networking. In *International Conference on Networking (IFIP Networking'13)*. IEEE, 1–9.
[113] Kai Wang, Jia Chen, Huachun Zhou, and Yajuan Qin. 2012. Content-centric networking: Effect of content caching on mitigating DoS attack. *International Journal of Computer Science Issues* 9, 6 (2012), 43–52.
[114] Kai Wang, Jia Chen, Huachun Zhou, Yajuan Qin, and Hongke Zhang. 2013. Modeling denial-of-service against pending interest table in named data networking. *International Journal of Communication Systems* 26 (2013), 1–14.
[115] Kai Wang, Huachun Zhou, Yajuan Qin, Jia Chen, and Hongke Zhang. 2013. Decoupling malicious interests from pending interest table to mitigate interest flooding attacks. In *IEEE International Workshop on Management of Emerging Networks and Services (Globecom)*. IEEE, 963–968.
[116] Yu Wang, Mingwei Xu, Zhen Feng, Qing Li, and Qi Li. 2014. Session-based access control in information-centric networks: Design and analyses. In *IEEE International Performance Computing and Communication Conference (IPCCC'14)*. IEEE, Austin TX, USA, 1–8.

[117] Walter Wong and Maurício Ferreira Magalhães. 2012. Security approaches for information-centric networking. In *Applied Cryptography and Network Security*. Springer, 76–98.

[118] Walter Wong and Pekka Nikander. 2010. Secure naming in information-centric networks. In *Re-Architecting the Internet Workshop*. ACM, 1–6.

[119] Christopher Wood and Ersin Uzun. 2014. Flexible end-to-end content security in CCN. In *Consumer Communications and Networking Conference (CCNC'14)*. IEEE, 1–8.

[120] Mengjun Xie, Indra Widjaja, and Haining Wang. 2012. Enhancing cache robustness for content-centric networking. In *International Conference on Computer Communications (INFOCOM'12)*. IEEE, 2426–2434.

[121] Zhiwei Xu, Bo Chen, Ninghan Wang, Yujun Zhang, and Zhongcheng Li. 2015. ELDA: Towards efficient and lightweight detection of cache pollution attacks in NDN. In *Conference on Local Computer Networks (LCN'15)*. IEEE, 1–9.

[122] Cheng Yi, Alexander Afanasyev, Ilya Moiseenko, Lan Wang, Beichuan Zhang, and Lixia Zhang. 2013. A case for stateful forwarding plane. *Computer Communication* 36, 7 (2013), 779–791.

[123] Yingdi Yu, Alexander Afanasyev, David Clark, K. C. Claffy, Van Jacobson, and Lixia Zhang. 2015. Schematizing trust in named data networking. In *ACM Conference on Information-Centric Networking (ICN'15)*. ACM, 177–186.

[124] Yingdi Yu, Alexander Afanasyev, Jan Seedorf, Zhiyi Zhang, and Lixia Zhang. 2017. NDN DeLorean: An authentication system for data archives in named data networking. In *ACM Conference on Information-Centric Networking (ICN'17)*. ACM, 11–21.

[125] Guoqiang Zhang, Yang Li, and Tao Lin. 2013. Caching in information centric networking: A survey. *Computer Networks* 57, 16 (2013), 3128–3141.

[126] Jianqing Zhang, Qinghua Li, and Eve Schooler. 2012. iHEMS: An information-centric approach to secure home energy management. In *International Conference on Smart Grid Communications (SmartGridComm'12)*. IEEE, 217–222.

[127] Lixia Zhang, Alexander Afanasyev, Jeffrey Burke, Van Jacobson, K. C. Claffy, Patrick Crowley, Christos Papadopoulos, Lan Wang, and Beichuan Zhang. 2014. Named data networking. *ACM SIGCOMM Computer Communication Review* 44, 3 (2014), 66–73.

[128] Lixia Zhang, Deborah Estrin, Jeffrey Burke, Van Jacobson, James Thornton, Diana Smetters, Beichuan Zhang, Gene Tsudik, K. C. Claffy, Dmitri Krioukov, Dan Massey, Christos Papadopoulos, Tarek Abdelzaher, Lan Wang, Patrick Crowley, and Edmund Yeh. 2010. *Named Data Networking (NDN) Project*. Technical Report NDN-0001. NDN Project.

[129] Xinwen Zhang, Katharine Chang, Huijun Xiong, Yonggang Wen, Guangyu Shi, and Guoqiang Wang. 2011. Towards name-based trust and security for content-centric network. In *International Conference on Network Protocols (ICNP'11)*. IEEE, 1–6.

[130] Zhenkai Zhu, Jeff Burke, Lixia Zhang, Paolo Gasti, Yanbin Lu, and Van Jacobson. 2011. A new approach to securing audio conference tools. In *Asian Internet Engineering Conference (AINTEC'11)*. ACM, 120–123.

Received June 2017; revised October 2018; accepted February 2019