Hey, I Know You Are Moving: Producer Movement Status Privacy in Information-Centric Networking

Qingji Zheng†, Qi Li‡, Aytac Azgin†, Syed Obaid Amin†
† Huawei Research Center, Santa Clara, CA, USA.
‡ Graduate School at Shenzhen, Tsinghua University, China.
†{qingji.zheng,aytac.azgin,obaid.amin}@ huawei.com
‡qi.li@sz.tsinghua.edu.cn

***Abstract*—Information-centric networking (ICN), an alternative for future networking architecture, offers great benefits to both consumers and producers from various dimensions, such as scalability, security and mobility. Unlike IP packets containing geographically dependent and host-centric IP addresses, neither ICN interest packets (issued by a content requester or consumer) nor ICN data packets (generated by a content source or producer) by-default leak any geographical information, and thus both consumers' and producers' locations are perfectly hidden *if the adversary does not eavesdrop on the network communication*. Location privacy, however, does not necessarily imply the privacy of movement status (i.e., whether the object is moving or not) that might open the door to profiling attacks towards producers especially in an IoT-driven era where each individual may act as a producer. In this paper, we are motivated by the following critical questions: (i) Can ICN preserve privacy of the producer's movement status, and (ii) If not, how can we preserve privacy of movement status if a producer is moving. We survey the current ICN mobility mechanisms and find that most of them are vulnerable to attaks on privacy of the movement status. We propose a simple approach that can be used to prevent such attacks. The objective of this paper is to raise the security awareness for network architects/designers when designing mobility support mechanisms.**

I. Introduction

Information-centric networking (ICN) has emerged as a strong candidate for the future Internet architecture. Among others, security enforcement and mobility support have been considered as key requirements when designing the ICN architecture. Many works, such as CCNx [1] and NDN [2], have been proposed by following some general design principals, including supporting security by design, decoupling content names from locations, stateful forwarding and pervasive caching.

Unlike the IP architecture, ICN provides a natural approach to support mobility and it preserves location privacy (i.e., without inferring one's, i.e., an end point's, location) for both consumers and producers. According to the design principal of decoupling content name from its location, data retrieval requests (i.e., interest packets) in ICN carry names of the desired content, and routers can use these names to route the interests, which will reach the producer if the interests are not satisfied by any intermediate router. In contrast to the IP architecture where the data is always associated with a host (i.e., producer), the consumer in ICN is only required to know the content name without caring about its location (or the producer's location). In other words, interest packets carry no information on producer location, which thus perfectly preserve the producers' location privacy [1]. In addition, in ICN as the requested content is sent back in reverse along the interest forwarding path, data packets also carry no location information about the consumer (as opposed to source IP in IP architecture).

While ICN naturally provides strong location privacy by design, in this paper we consider a different but relevant privacy issue –*movement status privacy*. The term "movement status privacy" means that one is attempting to preserve its movement status without being leaked to any potential adversary, no matter it is moving or not, and we say "break the movement status privacy" meaning that an adversary, having no privilege to eavesdrop on the ICN communication[2], can infer one's movement status. Here, "one is moving" means that one changes its location which might result in joining a different network from the previous one (i.e., leaving a network domain and joining another one) or change some .

In ICN although location privacy can be preserved perfectly (e.g., using location-independent routable names), movement status privacy may not be preserved as location privacy is not equal to movement status privacy. It is easy to see that preserving movement status privacy may imply preserving location privacy because if the location privacy is not preserved, then the adversary can easily use that location information to infer one's movement status. However, preserving location privacy does not imply preserving movement status privacy because an adversary may know one's movement status, but it may not be able to break the location privacy. People may argue that comparing with the location privacy which hides address information, movement status privacy only refers to one' movement status (i.e., either moving or static), which may not be a big issue. However, if movement status privacy is broken, it may impose great privacy risk: the adversary can use this piece of information as side channel to infer one's behavior pattern (e.g., by continually monitoring one's

[1]In the IP architecture, IP address is location-dependent and can thus be used to infer a bunch of location information with tools like WHOIS

[2]We assume that ICN is treated as a black box and the adversary cannot observe the data transmission within the network.

978-1-4673-9292-1/16/$31.00 ©2016 IEEE

movement status, the adversary can profile the associated behavior pattern).

Intuitively, movement status privacy in ICN can be separately considered with respect to the consumer and the producer. In this paper we focus on producer movement status privacy because consumer movement status privacy can be preserved naturally (see section II).

Our contribution. can be summarized as follows:

- We *first* consider the producer movement privacy and demonstrate the possible privacy breach via simulation. Specifically, we show that round trip time difference methodology (used in cache privacy attack [18], [3]) can be applied to launch producer movement status attack. We show that the adversary can exploit two simple approaches to launch privacy attacks without prior information about the producer.
- We survey the mobility mechanisms in the literature and find that neither of the existing approaches/mechanisms can preserve producer movement status privacy. We then propose a simple but effective approach to ensure the producer movement status privacy (to some extent).

In short, the main objective of this paper is to raise the awareness of ICN designers on privacy concerns regarding host mobility, with specific emphasis on the producer mobility. We believe that our work is timely and important because one of the ICN objectives is to provide security guarantees by design, which is different from the traditional IP architecture which provides enhanced security after design. Understanding the potential attack towards movement status privacy may help designers be aware of possible privacy leakages and therefore explore more effective approaches to mitigate that problem.

Organization. Section II presents an overview of the ICN. Section III describes the privacy attack(s) on producer movement status. Section IV shows a few possible approaches that the adversary can use/exploit to launch such privacy attacks. Section V surveys the current mobility mechanisms and presents a simple but effective countermeasure. Section VI briefly describes the related work and section VII concludes the paper.

II. ICN Overview

ICN communication is carried out by two types of packets: interest packet and data packet, which are uniquely identified by names. To be more specific, the name in the interest packet specifies the data to be fetched and the name in the data packet identifies the data it carries. Depending on the applications, names can be either hierarchically structured (e.g., /cnn/ca/sanjose/01.av) or flat (e.g., a GUID composed of the combination of the hash of producer's public key and sequence number). Note that while ICN names may indicate location information (e.g., /cnn/ca/sanjose/), they are not necessarily associated to any location information by using some approaches, such as flat name.

Roughly speaking, ICN routers are equipped with three components: (i)content store (CS), caching data packets that pass through the router. (ii) pending interest table (PIT), storing interest packets that have been forwarded (to upper link) but still waiting for matched data packets. and (iii) forwarding information base (FIB), storing the tuples of name prefix and outcoming interfaces. When the router receives an interest packet, it will conduct name match [3] to find the outcoming interfaces.

With the three components, the router handles the communication as follows:

- When an interest packet arrives, the router proceeds as follows: (i) If the requested data is cached by CS (i.e., via name match), then the router fetches the data and returns it to the consumer through the interface where the interest packet came in. Otherwise, it continues to next. (ii) If there exists an entry in PIT having the same name as the interest packet, the router appends the incoming interface of the Interest packet to that entry. Otherwise, it continues to next. (iii) If there exist outcoming interfaces in FIB with respect to the name in the Interest packet, the router forwards the interest packet to all outcoming interfaces. Otherwise, it aborts.
- When a data packet arrives, the router proceeds as follows: (i) If there is no entry in PIT having the same name as the data packet, the data packet will be dropped. Otherwise, it continues to next. (ii) The router forwards the data packet to all interfaces indicated by the PIT entry with the same name and then stores the data packet in CS.

We can see that routing interest packets towards producers is based on name (through FIB) and routing data packets is based on the router's forwarding state (through PIT). If the name in the interest packet does not imply any location information, then the overall communication will not leak any location information if the attacker is unable to eavesdrop the ICN communication. That is, ICN enjoys the desired property of preserving location privacy, comparing with IP architecture where IP address always implies the location information.

However, as argued in section I, location privacy does not imply movement status privacy. Specifically, we consider producer movement status privacy since consumer movement status privacy can be naturally hidden if the adversary has no capability to eavesdrop ICN communication even if the consumer is moving. The reason is that if the consumer moves before getting returned data packets, the consumer will detect the timeout and resend the interest packets, which will be satisfied by the intermediate router in high probability because of in-network cache. Therefore, the producer (if malicious) is unable to detect this event.

III. Producer Movement Status Attack

In this section, we describe the the privacy attack on producer movement status, demonstrate it through a simple simulation and then discuss its negative effect.

[3]Note that CCNx uses the exact match and NDN uses the longest prefix match. This is the significant difference between the two ICN instances.

A. Producer Movement Status Attack

Roughly speaking, producer movement status attack refers to a privacy attack where the adversary attempts to infer the producer's movement status by gathering additional side-channel information. We assume that the adversary (i) has no prior knowledge about the producer's movement status, (ii) cannot monitor/inspect the movement status directly and (iii) has no capability to eavesdrop on the communication between the network and the producer. In this paper, we consider the case where the adversary uses the round trip time (e.g., the time interval between issuing an interest packet and receiving the returning data packet), adopted by cache privacy attack [3], [18], to launch the producer movement status attack.

To be specific, the adversary (i.e., acting as a consumer) can infer the producer movement status by measuring the round trip time through continually requesting fresh data from the producer. Here the term "fresh data" means the data to be requested each time is different from each other and therefore the interest packets issued by the adversary can not be satisfied by any other host (e.g., an intermediate router). For example, the adversary issues an interest packet for fresh content D_1 and measures the round time trip τ_1 at time T_1. At time T_2 where $T_2 - T_1$ is a short interval (e.g., the necessary time interval for the producer to move out of current scope), the adversary issues another interest packet for another fresh content D_2 and thus can measure the round time trip τ_2. If there exists some significant difference[4] between the round trip times τ_1 and τ_2 (either $\tau_1 >> \tau_2$ or $\tau_1 << \tau_2$), the adversary may make the conjecture that the producer is moving. Otherwise, the adversary may conclude that the producer is static (meaning its location is within some scope).

Apparently, the intuition behind this attack is that when a producer is moving, the round trip times for successive data requests will change due to a change in routing path (e.g., routing path may become shorter or longer, meaning that the number of hops may change) between the producer and the adversary (who is static). Note that this attack does not require the adversary to have any extra capabilities such as hacking the network routers or eavesdropping ICN communication.

B. Attack Simulation

To demonstrate the producer movement status attack, we consider a simple simulation scenario for which topology is shown in Figure 1. The simulation setup consists of three different entities, which include consumer (acting as the adversary), producer (targeted entity) and ICN/content routers. We ran the simulations by changing the producer from co-ordinates (1,1) to (1, 2) and then (1, 3) and the link bandwidth from 1, 2, 4, 8 to 16 (Mbps), while keeping the consumer static. The consumer always issues interest packets toward the producer to request fresh data. We ran each scenario for 10 times to obtain the average round trip time.

[4]Note that even in the same location, the round trip time τ may be slightly different due to the network condition, such as network congestion.

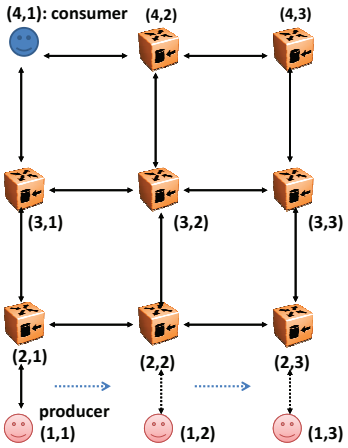

Fig. 1. Network topology of the simulation consists of 1 consumer, 8 routers and 1 producer. The network entities were organized as a grid starting with co-ordinates of (1,1) at the left-bottom. Each link delay is set to 10ms.

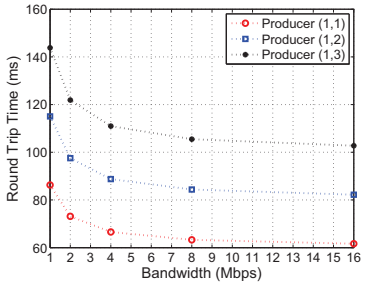

Fig. 2. Measured average round trip time by varying the producer's position from (1, 1) to (1, 2) and then (1,3). The bandwidth for all links are the same and varies from 1, 2, 4, 8 to 16 (Mbps). The size of requested data packet is 1024 bytes.

We show the simulation results in Figure 2. First, we can see that under the same network condition (i.e., the same link bandwidth), the round trip times show significant difference for different locations. For example, given the link bandwidth 1Mbps, the round trip time is around 90ms when the producer was at location (1, 1) while the round trip time is around 120 ms when the producer was at location (1, 2). Second, we observe that the round trip time decreases when the link bandwidth increases, and to some extent the round trip time is dominated by the link propagation delay rather than the link bandwidth (if the size of data packet is far less than bandwidth). Take the producer at location (1, 1) as an example, when the bandwidth is increased to 8Mbps or 16Mbps, the round trip times are almost close to 60ms, which equals the cumulative end-to-end link delay (i.e. up-link delay plus down-link delay). To sum up, from the given scenario we observe at least three factors that are essential to make producer movement status attack effective:

- The number of hops. When the forwarding path or route changes, it is possible that the number of hops along the path will change, which may vary the round trip time for each data retrieval.

- The link delay (i.e. link propagation delay depending on the medium that is used to transfer the packets, distance between routers and so on, and transmission capacity) along the routing path. Even if the number of hops remains the same for a different forwarding path, the link delay will vary because of available resources along the path (computing/bandwidth resources at the content routers) and the link utilization rates, which will ultimately vary the round trip time.

In other words, even when the producer is moving, movement status attack might not work if the hops along the forwarding path remains the same and the link delay do not change significantly. Unfortunately, the hop-count (between the static consumer and the moving producer) changes in high probability for scenarios involving mobile ad hoc networks [16].

C. Negative Effect of Producer Movement Status Attack

We argue that producer movement status attack might open the door for further attacks and influence all network users. First, producer movement status attack may enable the adversary to profile producers' behavior patterns. If the adversary is actively launching the movement status attack continually, then he can observe the movement status change, and then infer the victim's behavior patterns. Such behavior patterns might be valuable for the adversary to profile the producer who is quite sensitive for his own privacy, and the severity of such attack might depend on the adversary's capability and the specific target. In addition, as IoT has been pervasively utilized in various areas such as health-care system, each individual may play the role of the producer because wearing gadgets continually produces fresh data that will be collected/shared/stored by many applications. If producer movement status privacy can be exploited, then movement status of the person wearing that device can be explored.

IV. How to Launch Attack

We can see that launching movement status attack requires the adversary (who acts as a consumer) to measure the round trip time between himself and the producer. Therefore, the natural problem is that how the adversary can measure the round trip time without any prior knowledge about producer and without compromising network facilities. That is, the adversary should guarantee what it requests will directly be sent by the producer, rather than by some intermediate router. For CCNx, the most simple approach is to exploit the "AnswerOriginKind' field' in the interest packet by setting the low order bits to zero, which explicitly states that "do-not-answer-from-content-store" [5]. In what follows, we present two general approaches that enable the adversary to achieve this goal without any restriction on a specific ICN implementation.

A. Requesting Fresh Data

As mentioned earlier, one approach that the adversary can use is to request fresh (or real-time) data, which is not cached by any intermediate router. For example, if the producer is wearing the smart device that always generates time-sensitive data, then the adversary can send interest packets to retrieve it. Doing so guarantees the interest packets to ultimately reach the producer.

While this approach seems quite intuitive and easy for the adversary, it does require the producer to always feed real-time data to the network and the adversary to have some sort of prior information about the producer (e.g., the producer is wearing a smart device). This approach might not work if the producer only shares/generates either stale or archival data.

B. Requesting No-Existing Data

In ICN negative acknowledgments (NACKs) are introduced as an alternative to notify the consumer that the content it requests does not exist. Introducing NACKs does bring some advantages. With NACKs, a consumer can quickly identify non-existing data without waiting for the issued interests to time out. Moreover, using NACKs enables consumers to differentiate the cases of non-existing data and interest packet loss.

However, using NACKs will also facilitate the adversary to exploit the producer movement status without requiring any prior knowledge about the producer. To be specific, the adversary can continually issue the interest packets whose names are composed of the producer prefix and some random numbers, such as /producer_prefix/random_number. The purpose of using "random_number" is two-fold: (i) it assures that the interest packet will never be satisfied by an intermediate router, and (ii) it assures that the producer does not possess the requested data and will reply with the NACKs. By doing so, the adversary can evaluate the round trip time successfully without any prior knowledge except the producer prefix.

C. Enhance Attack with Neighborhood Reference Information

While measuring the round trip time may help the adversary to infer whether the producer is moving or not, it may be insufficient to make a decision on its movement because of unpredictable/undetermined network conditions. For example, a sudden incurring network congestion may increase the round trip time even the producer is static. Therefore, in order to improve the success rate of such attack, the adversary might utilize neighborhood information as reference. For example, if the adversary knows some related fixed producer residing in the same scope with the target producer, then he can also keep probing the round trip time with respect to the fixed producer. Since the fixed producer and the target producer shares majority part of the network, this reference information (e.g. the round trip time with respect to the fixed producer) can greatly help the adversary to infer the target producer's movement status.

V. ICN Mobility Mechanisms and Countermeasure

In this section, we survey the current ICN mobility mechanisms to see whether these mobility mechanisms can preserve the producer movement status privacy or not.

[5]https://www.ccnx.org/releases/latest/doc/technical /InterestMessage.html

A. ICN Mobility Mechansims

Agent-based Mobility Mechanisms Agent-based mobility mechanisms use a static home agent to track the moving producer. To be specific, the mobile producer always reports its new (or temporary) routable name to the static home agent, who later can use such information to guide interest packets towards the moving producer. There are two alternatives according to the way of static home agent handling the incoming interest packets.

- Forwarding [19], [13], [15]: The static home agent will forward the incoming interest packets to the producer by appending new routable name (e.g. encapsulating interest packets), so that the producer can receive interest packets and response accordingly. While this approach is simple from the implementation perspective, as it requires no modification to the current ICN architecture, it does bring forth a critical performance issue: the path of transmitting interest packets from the consumer to the producer is not always optimal because interest packets always need to pass through the home/producer agent. We also note that this approach might be vulnerable to the producer movement status attack since the adversary can easily infer the producer's movement status because the number of hops in the forwarding path will change and the measured round trip times will show some significant difference in high probability.
- Notification [19]: When the static home agent receives interest packets towards the moving producer, it will notify the customer with the new routable name of the moving producer. To this end, the consumer can re-send or re-issue the interest packets which can be routed to the moving producer. While this approach can achieve path optimization due to sending the request directly to the producer, apparently it cannot preserve producer's movement status privacy since the consumer is always notified with the new routable name.

Resolution-based Mobility Mechanism Resolution-based mobility mechanism adopts another extra component – resolution system (similar to DNS) – to help locate the moving producer. Specifically, it assumes that there exists a global name-locator resolution system (which can be either centralized or decentralized) storing the mapping between the moving producer's name and its locator. The moving producer's name/prefix can be mapped to its real-time locator by querying the resolution system, and therefore the interest towards the producer can be directly forwarded via locator. Depending on the manner of handling interest packets, there are two approaches to achieve the mobility as follows:

- Encapsulation: Given an interest packet towards the producer, the router at the edge will query the name resolution system to fetch the producer's locator. Given the locator, the router will encapsulate the original interest packets with the corresponding locator, so that interest packets will be routed accordingly. The drawback of this approach is that it cannot utilize the cache store effectively because different locators (i.e., name with respect to interest packets) might correspond to the same data but the router does not aware of this.
- Forwarding Label [5]: Given an interest packet towards the producer, the router at the edge will query the name resolution system to acquire the producer's locator, and then insert the locator in the field of "forwarding label". The intermediate router will check that field to decide the next hop it will route. Compared with the encapsulation approach, this approach can take advantage of in-network caching since requested prefix within the interest packet remains the same.

A resolution-based mechanism, without eavesdropping on the network communication, while the adversary has no idea whether the interest packet has been encapsulated (or appended with forwarding label) or not, can not guarantee that the round trip times will not show significant variations as the forwarding path might change due to producer movement. That is, resolution-based mechanism might also be vulnerable to the producer movement status attack.

Summary. We can see that current mobility mechanisms suffer from the producer movement status attack. The possible reason is that privacy (or security) might not be fully taken into account since functionality (i.e., how to support mobility from the perspective of efficiency and scalability) is the top priority when designing mobility mechanisms.

B. Countermeasure for Agent-based Mobility Mechanisms

We present a simple approach for agent-based mobility mechanism to preserve producer movement status privacy.

Agent as the Cache We note that producer movement status attack is to detect whether the producer is moving. Hence, if the producer is static (or always successfully pretending to be static), then its movement status privacy can be naturally preserved. In other words, if the producer can guarantee that the round trip time per each data retrieval shows no significant difference, then there is no need to preserve the movement status privacy anymore. While this idea is ideal, it is not realistic since producers have no control on the network (e.g., determining the routing path).

As an alternative, we propose an approach –"Agent as the Cache", which can help the agent-based mobility mechanism to mitigate the producer movement status attack. The basic idea is that the home agent with respect to the moving producer, always perform all functions on behalf of the producer, and can be treated as a data repository for the producer (for data retrieval). For each piece of data to be published, the producer will store then in the agent first and then delegate the agent the right to distribute it upon data request. If the requested data content does not exist in the repository, then the agent will return the NACK without querying the producer.

We can see that the approach of "Agent as the Cache" works well without imposing any complexity to the current ICN architecture, but enjoying the advantages as below: (i) the approach preserves producer movement privacy because the routing path between the producer and the consumer

is now replaced by that between the static agent and the consumer, meaning that if the round trip time always is approximately similar even the producer is moving. (ii) The approach does not require any change on the underlining ICN infrastructure and current agent-based mobility mechanisms, since the proposed approach only imposes modifications at the agent side.

However, this approach is not perfect since it requires that the producer stores all its data in the agent and might be not applicable for some applications. For example, some applications might require the producer to publish data depending on real-time decision (e.g., authenticating the consumer and then derive data from the user profile) and the producer is not allowed to delegate full functionality to the agent due to security concerns.

Obufuscation with Random Delay The "Agent as the Cache" approach attempts to obfuscate the producer's movement which is to be static. It is natural to think about the other approach where the producer responses the data requests with random delay in order to pretend to be always moving (no matter it is static or moving). That is, if the producer can pretend to be moving all the time, then it is impossible for the adversary to infer any useful information. We leave it as the further study.

VI. Related work

We briefly review the relevant security topics in ICN as follows.

Data-related security. ICN is based on the "data-centric" paradigm. In order to secure data in the ICN, many works have been proposed to achieve the objectives of data privacy data integrity and data provenance [14], [17], [12], [10], [11]. Note that these works are adopting the cryptographic techniques, such as encryption and signature, to secure the data.

Interest security. As the ICN communication is "pull-based" and initiated by interest packets, many attacks have been launched through issuing falsely interests, such as interest flooding [8], [4], cache poison [6] and cache pollution [9].

Protection against side-channel attacks. Many attacks have been implemented to infer entities' (either consumer, producer or application) behavior by using side channel information. For example, caching data in the ICN routers enables the adversary to infer nearby consumers' access behaviors by using the round trip time, and [18], [3] explored possible approaches to protect such privacy leakage. In addition, caching might enable the adversary to identify the consumer's location, and [7] presents many algorithms to demonstrate that by controlling many hosts, the adversary can leverage in-network caching mechanism (used as side channel) to locate consumers.

Note that the attack consider in this paper also belongs to this catalog, and we demonstrate that the side channel information can be utilized to infer producers' moving status, which has not been considered before.

VII. Conclusion

In this paper, we consider the attack against producer movement status, which explores the side channel information (round trip time in this paper) to infer the producer's moving status. We demonstrate the attack via simulation and we investigate some approaches that might be used by the adversary to achieve the attack. In addition, we survey the ICN mobility mechanism and we found that most are vulnerable to this attacks.

Acknowledgments.

Qingji Zheng is supported in part by NSFC under Grant No.61472472 and Qi Li supproted by NSFC under Grant No.61572278.

References

[1] Ccnx content centric networking.
[2] Named data networking (ndn) - a future internet architecture.
[3] G. Ács, M. Conti, P. Gasti, C. Ghali, and G. Tsudik. Cache privacy in named-data networking. In *IEEE ICDCS 2013, 8-11 July, 2013, Philadelphia, Pennsylvania, USA*, pages 41–51, 2013.
[4] A. Afanasyev, P. Mahadevan, I. Moiseenko, E. Uzun, and L. Zhang. Interest flooding attack and countermeasures in named data networking. In *IFIP Networking Conference, 2013, Brooklyn, New York, USA, 22-24 May, 2013*, pages 1–9, 2013.
[5] A. Azgin, R. Ravindran, and G. Wang. Mobility study for named data networking in wireless access networks. In *IEEE ICC 2014, Sydney, Australia, June 10-14, 2014*, pages 3252–3257, 2014.
[6] R. Bassil, R. Hobeica, W. Itani, C. Ghali, A. I. Kayssi, and A. Chehab. Security analysis and solution for thwarting cache poisoning attacks in the domain name system. In *19th International Conference on Telecommunications, ICT 2012, Jounieh, Lebanon, April 23-25, 2012*, pages 1–6, 2012.
[7] A. Compagno, M. Conti, P. Gasti, L. V. Mancini, and G. Tsudik. Violating consumer anonymity: Geo-locating nodes in named data networking. 2015.
[8] A. Compagno, M. Conti, P. Gasti, and G. Tsudik. Poseidon: Mitigating interest flooding ddos attacks in named data networking. In *IEEE LCN'2013, Sydney, Australia, October 21-24, 2013*, pages 630–638, 2013.
[9] M. Conti, P. Gasti, and M. Teoli. A lightweight mechanism for detection of cache pollution attacks in named data networking. 2013.
[10] N. Fotiou, G. F. Marias, and G. C. Polyzos. Access control enforcement delegation for information-centric networking architectures. *Computer Communication Review*, 42(4):497–502, 2012.
[11] C. Ghali, G. Tsudik, and E. Uzun. Elements of trust in named-data networking. *CoRR*, abs/1402.3332, 2014.
[12] A. Ghodsi, S. Shenker, T. Koponen, A. Singla, B. Raghavan, and J. Wilcox. Information-centric networking: seeing the forest for the trees. In *HOTNETS '11, Cambridge, MA, USA - November 14 - 15, 2011*, page 1, 2011.
[13] F. Hermans, E. Ngai, and P. Gunningberg. Global source mobility in the content-centric networking architecture. NoM '12, pages 13–18, 2012.
[14] M. Ion, J. Zhang, and E. M. Schooler. Toward content-centric privacy in ICN: attribute-based encryption and routing. In *ICN'13, August 12, 2013, Hong Kong, China*, pages 39–40, 2013.
[15] J. Lee, S. Cho, and D. Kim. Device mobility management in content-centric networking. *IEEE Communications Magazine*, 50(12):28–34, 2012.
[16] S. Merkel, S. Mostaghim, and H. Schmeck. A study of mobility in ad hoc networks and its effects on a hop count based distance estimation. In *NTMS 2012, May 7-10, 2012*, pages 1–5, 2012.
[17] S. Misra, R. Tourani, and N. E. Majd. Secure content delivery in information-centric networks: design, implementation, and analyses. In *ICN'13, August 12, 2013, Hong Kong, China*, pages 73–78, 2013.
[18] A. Mohaisen, X. Zhang, M. Schuchard, H. Xie, and Y. Kim. Protecting access privacy of cached contents in information centric networks. CCS '12, pages 1001–1003, New York, NY, USA, 2012. ACM.
[19] Y. Zhang, H. Zhang, and L. Zhang. Kite: A mobility support scheme for ndn. ICN '14, pages 179–180, New York, NY, USA, 2014. ACM.