A Survey of Security Problems in Content Centric Networking

Yicheng Yang[1], Haohao Kang[1], Jianlong Yang[2], Huanyu Wu[1], Yi Zhu[1*]

[1]School of Computer Science and Communication Engineering,
Jiangsu University, Zhenjiang, China
[2]Jingjiang College, Jiangsu University, Zhenjiang, China

Abstract: Today, content centric networking (CCN) has been a research hotspot of future Internet architecture. Adopting a distributed content caching mechanism, CCN works in content oriented mode instead of the traditional host-to-host delivery mode. It can effectively shorten the data fetching delay and control network congestion. But the research on CCN security is still at the preliminary stage. Although CCN can solve some traditional network attacks due to its security mechanisms design, some new security threatens also emerge simultaneously. Today, cache pollution, PIT flooding, cache privacy leak, content poisoning and content privacy leak have become serious security risks in CCN. Based on the introduction of working mechanism, this paper discussed the cause, damage and countermeasures of five important security problems above. This paper can provide valuable reference to future research in the security area of CCN.

Keywords: Content centric networking; Security threaten; Cache pollution; PIT flooding; Privacy leak; Content poisoning

1 Introduction

With the rapid development of Internet, the aims of people are becoming clear toward to network contents, and the network application body has also been gradually shifting to content services. Unfortunately, the traditional IP network architecture is based on host-to-host mode, so it cannot satisfy current Internet development requirements. In order to solve this problem, European and American researchers have started several research projects about next generation Internet architecture since 2006, including DONA[1](Data-Oriented Network Architecture) proposed by UC Berkeley RAD lab, 4WARD[2] by European Union FP7, PSIRP[3] (The Publish-Subscribe Internet Routing Paradigm) and CCN[4][5](Content-Centric Networking) by Palo Alto Research Center, and NDN (Named Data Networking) by NSF Future Internet Architecture (FIA). Without exceptions, these projects all adopt content centric idea to design network architecture. So today, CCN has become a representative and research hotspot of next generation Internet.

* Corresponding author: Yi Zhu (zhuyi@ujs.edu.cn).

Compare with traditional network, CCN has already considered a certain security mechanism in its original design. Unlike IP network try to protect the security of link connection, CCN aim at protecting content itself. In CCN, all contents must be authenticated by digital signature and encrypted before dissemination. Furthermore, because CCN use content name for routing in place of host address, attacker cannot launch an attack to specific CCN node just like DDoS (Distributed Denial of Service) in IP network. But although CCN has already solved part security problems in traditional network, several new threats are arising simultaneously. Currently, CCN is confronted with five important security threats—cache pollution attack, content poisoning，PIT flooding attack，cache privacy leak and name privay leak.

This paper focuses the above security problems. Based on the introduction of working mechanism, the causes, damages and countermeasures of these five security threats are further discussed. We think this paper can provide a certain theoretical reference for security research in CCN.

2 Working Mechanism of CCN

CCN uses content name as identification for routing instead of IP address in the current IP networks. Hierarchical naming mechanism similar to URL is employed, e.g. “ujs.edu.cn /Video/ Lecture_1.mpeg”, where “/ujst.edu.cn/Video” is the content prefix used for content retrieving and forwarding, “/ujs.edu.cn” represents the content provider, “Video/“represents the content type and “/Lecture_1.mpeg” represents the content itself.

There are two kinds of packets in CCN: interest and data. Interest packets contain content identification, selector, and nonce. The selector comprises order preference, publisher filter, and scope. Data packets contain signature, signed info, key locator and stale time, and content. The signature comprises digest algorithm and witness, and the signed info comprises a publisher ID.

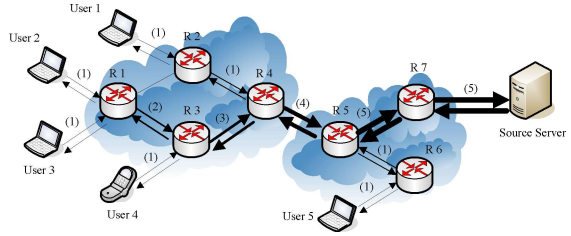
The key structure of a CCN node/router is composed of a Content Store (CS), a Pending Interest Table (PIT) and a Forwarding Information Base (FIB). CS provides storage space for caching contents. PIT records the received Interest Packets with their arriving faces, which are being pended for response. FIB indicates the next hop to forward the Interest Packets. The requested contents will be cached as much as possible in network, they can be quickly provided if other users request the same contents subsequently. This is a completely different to the way a traditional IP router works. Usually, a traditional IP router clears the cache on forwarding.

Operation of maximum matching query is executed on CS, PIT and FIB in turn when an Interest Packet arrives at the node. If the requested content is found in CS, it will be sent to the requester through the arrival face of Interest Packet. If the requested content is not found in the CS, the PIT is queried. If the PIT contains the related content entry, the PIT indicates that the content request has been received and waits for response. In doing so, it adds the arrival face the content’s entry; otherwise, the FIB is queried further on. If the FIB has the related content entry, the interest packet is forwarded

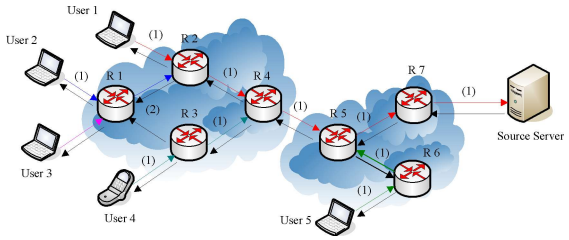
 ISSN 2572-4975 (Print), 2572-4991 (Online)

through the face indicated by the FIB. If no match is found in the FIB, the interest packet is dropped.

To make a comparison, the process of content delivery in IP network and CCN is shown in Fig.1. With the IP client/server infrastructure, each piece of content delivered has a round trip from the request user to the source server. A request that involves a large amount of content involves a huge amount of network traffic that is likely to cause network congestion or server overload. With the CCN infrastructure, the user may obtain the content from the cache of a nearby node. This eliminates traffic further along the line to and from the source severs. In Fig. 1, the request from user 1 goes to the source server (as in a conventional IP network). However, the content can be cached in routers R2, R4, R5 and R7 on its way back to user 1. If user 2 subsequently requests the same content, R2 can deliver it because there is a content copy in its cache. Similarly, the content can be cached in R1 on the way to user 2. When user 3 requests the same content, they can simply get it from the neighboring router R1. This only involves one hop. Through the caching mechanism and content identification that is independent of location, terminal users can obtain content from the network node that is as near to the user as possible.



(a) IP-based network infrastructure



(b) CCN-based network infrastructure

Fig. 1. Comparison of IP network and CCN

3 Main Security Threat Problems

Although CCN has many advantages, it also suffers some security threats. This section mainly describes the most important five security attacks in CCN.

3.1 Cache Pollution Attack

Cache pollution attacks mainly focus on the caching mechanism, which is the core characteristic of CCN. This attack includes two possible attack behaviors: ①Locality-Disruption[6], the attackers continuously generate a large number of illegitimate requests for new unpopular contents, thus disrupting the content locality of a cache; ② False-Locality[7], the attackers repeatedly request a certain popularity class contents so that this class contents will occupy cache space for long time, thus creating a false locality in a cache. False-Locality is easily confused with Flash-Crowd. Flash-Crowd is a phenomenon that hot contents such as breaking news are requested by lots of people, and hot contents quickly occupy the cache space. This feature is similar to False-Locality, so it is harder to identify False-Locality than Locality-Disruption. Obviously, cache pollution attacks aim at artificially decreasing the caching proportion of high popularity contents in cache, and then decrease the request hit probability of CCN node, increase content visiting delay, and damage the network performance of CCN. These two class attack behaviors mentioned above, the former smooths the arrival requests distribution of CCN node, the latter sharpens the arrival requests distribution of CCN node. Because caching mechanism determine the CCN's running efficiency, cache pollution attacks do great damage to CCN.

3.2 PIT Flooding Attack

According to the original design of CCN, CCN router must record the forward status of unsatisfied interest packets in PIT. But only the interest packets are satisfied in network or overtime, the entries in PIT can be deleted. This feature of maintaining the interest packets forward status is easily used by attackers due to the limited computing resource of router and limited storage space of PIT [8]. When attackers send a large amount of illegitimate interest packets to a router, the storage space of PIT in this victim will rapidly exhausted, so that router cannot create new PIT entry to record legitimate user's interest packet and its arrival port, and then result network congestion. This attack method is usually called as PIT flooding, just like DDoS attack in IP network.

In order to ensure the effect of PIT flooding and exhaust router's resource as much as possible, attackers must try to avoid flooding the similar name interest packets and avoid the requested contents are satisfied in network [9]. If illegitimate interest packets refer to the existing data packets, attackers must collect a lot of unpopular contents' name. It increases the attack cost while the attack effect will be bad. So PIT flooding attackers are more inclined to fake interest packets which refer to some non-existent contents. Because CCN router cannot judge the authenticity of received interest packets, these faked illegitimate interest packets will be stored in PIT until timeout.

3.3 Cache Privacy Leak

Every CCN node all has a Content Store (CS) which is used for caching data packets. Through content name addressing mechanism, users can obtain data packets from

 ISSN 2572-4975 (Print), 2572-4991 (Online)

nearby router's cache [10]. This mechanism can accelerate network response time, mitigate network congestion and promote the utilization rate of network resources. But as an open data exchange platform, caching mechanism also causes the content privacy leak problem while increasing network performance [11]. Using cache probing method, attackers can trace the visiting process for sensitive contents of neighbor users, and then reach the targets of snooping neighbors' privacy or analyzing neighbors' behaviors.

Content retrieving time measure is the main method used by attackers. As shown in Fig.2, we assume that attacker (U2) and legitimate user are neighborhood, and they exist within the scope of access router R1. U2 can deduce whether U1 requests specific content recently by measuring the round-trip time (RTT) of this specific content. The detail measuring process as follow: First attacker requests any content (not existing in network) from source server, then measures the round-trip time from source server, we define this delay as RTT_S. Second, attacker further measures the round-trip time from closest router (just as R1 in Fig.2), we define this delay as RTT_C. After above preparation, attacker begins to probe the target content, and we define the fetching time is RTT_A, if

(1) $|RTT_A - RTT_C| < \varepsilon$ （$\varepsilon \to 0$）, obviously the target content has already existed in closest router, attacker can deduce his/her neighbor user requested this target recently. Note that "recently" in this case means the cache update time of CCN router.

(2) $RTT_A > RTT_C$ and $RTT_A < RTT_S$, this target does not exist in the closest router but exists in the network, attacker can deduce the neighbor user requested this target in the past long time, but did not request it recently.

(3) $|RTT_A - RTT_S| < \varepsilon$,this target should be fetched from source server, attacker deduces that the neighbor user didn't request it in the past long time.

The above attack can effectively probe the specific content request behaviors of neighbor user within one hop scope. Even if attacker has many neighborhoods, attacker can also obtain the privacy information of victim if attacker has learned some prior knowledge.

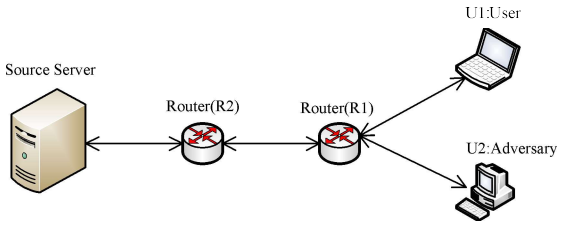


Fig. 2. Typical Scenario of Cache Privacy Leak

3.4 Content Poisoning

Content poisoning is an important issue in CCN. In this attack, an attacker uses legitimate names to inject spurious content into network [12]. In other words, the name of released content is legal, but the content is illegal. Although users can refuse the content

for the reason of failed signature verification, content verification needs key retrievals, which is a high overhead to the router. So, subsequent requests may be satisfied by illegal contents continuously and the overhead of routers will be increasing. In paper [13], Gasti and Paolo pointed out that if the size of packets is 1.5KB, even if the optimized RSA algorithm is used, the max bandwidth of router with Intel Core 2 Duo 2.53GHz CPU is about 150Mbps. Content poisoning is mainly a forwarding problem. Malicious contents are allowed to be filled into network, and routers forward requests from users to malicious content producers. This is different from cache pollution, because cache pollution returns legal contents which are needed by users, but content poisoning returns illegal contents.

3.5 Name Privacy Leak

Name privacy refers to the privacy leak of content names in CCN. Although CCN utilizes digital signature authentication method, which can encrypt the contents produced, CCN utilizes content names for routing, and content names are often in plaintexts. Generally, names have a semantic connection with their contents. If a content name has more obvious meaning, more information of contents will be leaked. The attacker can use content names to infer sensitive contents in the cache, and then achieve private information by means of deep packet inspection.

4 Countermeasures

4.1 Countermeasures of Cache Pollution

To defense cache pollution, probabilistic caching strategy and traffic control mechanism of specific face are two typical methods.

4.1.1 Design Cache Strategy

In terms of probabilistic cache strategy, Xie et al. [14] introduced CacheShield in his paper, and it is effective to fight against Locality-Disruption. The key metric of CacheShield is shown in equation (1)

$$\psi(t) = \frac{1}{1+e^{(p-t)/q}}, t = 1, 2, ... \tag{1}$$

where $\psi(t)$ represents the caching probability of t^{th} request, p is the average number of requests for the specified category content in a statistical period, q is the adjustment parameter. According to equation (1), the probability of contents stored in CS is low if t is less than p; if t is equal to p, the probability of caching is 50%; when $t>p$, with the increase of requests, the probability will gradually increase, and finally tend to 1. Obviously, this mechanism can effectively suppress CS to cache unpopular contents, therefore it can prevent Locality-Disruption, but it cannot defend against False-Locality. In the other side, CacheShield must be running continuously. Even no cache pollution occurs, it still causes overhead in routers.

 ISSN 2572-4975 (Print), 2572-4991 (Online)

4.1.2 Face Traffic Control

In paper [15], hit rate of face is the main indicator to detect cache pollution attack. If Locality-Disruption occurs, the face hit rate will decrease; if False-Locality occurs, the face hit rate will be inflated. Therefore, by detecting whether the hit rate of face exceeds the threshold, the router can determine whether a Locality-Disruption or False-Locality attack occurs. When the former occurs, the router can limit the request rate of the face as the threshold of request rate multiplied by the face hit rate. When the latter occurs, the router rejects packets returned by the cache (i.e. sets the caching probability of the face to zero).

4.2 Countermeasures of PIT Flooding Attack

According to recent researches, attack detection and countermeasure in general is based on face traffic, this mechanism is also divided into discard mechanism, acceptance mechanism and retreat mechanism.

4.2.1 Interest Packet Discarding Mechanism Based on Face Fairness

Alexander et al. [16] proposed a token bucket algorithm based on the fairness of each face. This mechanism is essentially a queuing algorithm. Unlike normal queuing, queues of interest packets do not actually store packets, but bi-directional pointers to existing PIT entries. Therefore, PIT entries can be quickly updated when interest packets are forwarded, and can be easily removed from the queue when the interest packets expire. By setting the appropriate queue size, the authors can control the overhead of router. It is very important to set the threshold of queuing time of interest packets. If interest packets have been in the queue for a long time and expire, the data packet will be discarded at downstream routers.

However, the key disadvantage of this mechanism is that it still allows many malicious interest packets from attackers to pass. A large portion of these malicious interest packages will be forwarded to the content producer, thus reducing the resources available to serve legitimate users. This algorithm tries to ensure that each face does not forward interest packets beyond its fairness setting, but it will discard both legitimate and malicious interest packets.

4.2.2 Receiving and Fallback Mechanism Based on Interest Packet Satisfaction Rate

In [13], two methods were proposed. The first method is interest packet reception mechanism based on satisfaction. The face satisfaction is defined as the ratio of the number of successful returned packets to the number of requested interest packets in a statistic time. Obviously, face satisfaction is directly related to the severity of PIT flooding. The more severe the PIT flooding attack is, the lower the corresponding face satisfaction is. After the router successfully gets the statistics of interest packet satisfied rate, it can use this data to limit the malicious interest packet traffic. The rate of face after traffic limitation is equal to its original rate multiplied by the face satisfaction rate. The second method is fallback mechanism based on face satisfaction. In this method,

the arrival interest packets are restricted for each incoming face, where the restriction threshold value is directly dependent on the satisfaction rate of each face. Unlike the first method, the router declares the limitation threshold to its downstream neighbor routers. If the satisfied rate of interest traffic is below the threshold, router will limit the traffic of interest packet, then forward the warning message to its downstream router and execute restriction rule in downstream router.

The disadvantage of first method is that each router on the path makes independent decisions on forwarding or deleting interest packets, lacks mutual feedback. As a result of independent decision, the probability of forwarding legitimate interest packets decreases rapidly as the number of hops between users and content producers increases, which results in a decrease in the value of interest packet satisfaction. This result further restricts legitimate interest packets through the router, and causes a vicious cycle. The second method overcomes the shortcomings of the former two methods, avoids the one-size-fits-all decision through reasonable restriction criteria, and avoids the decision error caused by single router.

4.3 Countermeasures of Cache Privacy

Acs et al. [17] investigated cache privacy in named data network in the presence of timing and cache detection attacks. The authors demonstrate the effectiveness of these attacks in different network topologies, and the attack rate can reach 59% even when the attacker and the victim have three hops from the shared router. They discussed two types of traffic, interactive traffic and content distribution. In terms of interactive content, the author suggests attaching a random number to the content name which is mutually agreed by users and producers, and the method can prevent an attacker from successfully getting the content by requesting the exact content name. However, the disadvantage is that even if the content exists in cache, it cannot be satisfied by another user because of not adding the random number when requesting. This condition will decrease the performance of caching. In terms of content distribution, a router may add a manually set delay before returning privacy-sensitive content to hide the true latency. This strategy preserves one of the important benefits of router caching, namely, reducing congestion and saving bandwidth. However, if the delay value is unreasonable, the router may lose the advantage of rapid response to users.

In paper [18], Chaabane et al. suggest a method of delaying all requests or delaying the initial k requests to fight against timing attack. The delay is set as the round trip time(RTT) from users to the content producer. The router will return contents after delaying an RTT or k requests. They also briefly discuss another two methods called collaborative caching and random caching, to protect cache privacy. Collaborative caching increases anonymity of cached contents by caching them in a group of routers. In random caching, routers cache contents based on its location on the forwarding path and the available space in the cache. Because the attacker cannot know the exact decision of routers, this method can protect cache privacy.

Lauinger et.al. proposed the idea of selective caching [19], in which a content is cached only when the popularity reaches a certain threshold. If the popularity is too

 ISSN 2572-4975 (Print), 2572-4991 (Online)

low, the router will not cache it. This method is based on the view that non-popular contents have more privacy risks.

4.4 Countermeasures of Content Poisoning

The main reason for content poisoning is invalid validation mechanism and imperfect routing. In order to combat content poisoning, we must redesign the validation mechanism, and design a reasonable routing mechanism.

4.4.1 Content Verification Mechanism Design

Current CCN content verification mechanism exploration is still in its infancy, two main methods are as follows.

(1) Probability check. Bianchi et al. proposed to reduce content verification by decreasing cache probabilities. Although this idea has some advantages, but the author did not consider the relationship between processing capacity of nodes and traffic of content verification, and how to control repeat verification.

(2) Check on hit. Kim et al. [20] proposed a lightweight content verification scheme which can save a lot of computational overhead. After the content is cached, it will not be validated until the content is hit in CS. In order to save the verification overhead, the author uses the least recent permutation of the segments. However, the underlying forwarding problem is not solved and will cause poisoned content to be re-requested.

4.4.2 Improving Routing Mechanisms

In [21], DiBenedett et al. proposed two routing strategies to suppress content poisoning. One is Immediate Failover, it makes next hop routers which returned poisoned contents become the last choices for subsequent interest packets. Another is Probe First, it stops forwarding interest packets within the attacked namespace, and then probes all next hops by verifying returned packets. After successful authentication, it will recover normal forwarding to next hop. Using above strategies, most of interest packets can be forwarded to legal content producers, and the impact of content poisoning can be curbed from the source. However, the disadvantage of the methods is lack of effective coordination with the validation mechanism.

4.5 Countermeasures of Name Privacy

The naming method based on cryptographic hash in CCN was proposed by Baugher et.al. [22]. The main advantage of this self-verified name (i.e., the name is an encrypted hash value of content) is reducing the overhead of validation. In this scenario, the readable name of content is mapped to a directory with its hash value, through which the user can obtain the self-verified name of the content. Users store hash names for subsequent same requests, and then request contents corresponding with the names. If the encrypted hash value of retrieved content matches the self-verified name in the directory, it accepts the retrieved content. This mechanism can also be used to protect the

privacy of content producer. The authors point out that hash-based naming is only useful for read-only and cacheable content. However, using a directory to obtain a self-verified name requires a trust mechanism between user and the directory producer, which creates a trusted infrastructure foundation in network.

5 Conclusions

As one important solution of next generation Internet architecture, CCN has attracted the attentions of many researchers and enterprises. There are still some problems waiting for solving in CCN, such as routing design, mobility management and security, etc. In this paper, we only introduce the basic working mechanism of CCN, briefly describe its security problems, and discuss several countermeasures of these problems. As a new emerging network, more research works need to do for CCN in future.

Acknowledgments

This research is financially supported by the Jiangsu Province Innovation Training Program of University Student under Grant No. 201713986002Y and Innovation Practice Fund of Industry Center of Jiangsu University under Grant No. ZXJG201659.

References

1. Koponen, Teemu, et al., ***A data-oriented (and beyond) network architecture***, *ACM SIGCOMM Computer Communication Review*, Vol. 37, No. 4, ACM, (2007).
2. Ahlgren, B., et al., ***Second netinf architecture description***, *4WARD EU FP7 Project, Deliverable D-6.2 v2. 0* (2010).
3. Dimitrov, Vladimir, Ventzislav Koptchev. ***PSIRP project--publish-subscribe internet routing paradigm: new ideas for future internet***, *Proceedings of the 11th International Conference on Computer Systems and Technologies and Workshop for PhD Students in Computing on International Conference on Computer Systems and Technologies*, ACM, (2010).
4. Jacobson, Van, et al., ***Networking named content***, *Proceedings of the 5th international conference on Emerging networking experiments and technologies*, ACM, (2009).
5. Choi, Jaeyoung, et al., ***A survey on content-oriented networking for efficient content delivery***, *IEEE Communications Magazine* 49.3 (2011).
6. Conti, Mauro, Paolo Gasti, Marco Teoli, ***A lightweight mechanism for detection of cache pollution attacks in named data networking***, *Computer Networks* 57.16 (2013): 3178-3191.
7. Park, Hyundo, Indra Widjaja, Heejo Lee, ***Detection of cache pollution attacks using randomness checks***, *Communications (ICC), 2012 IEEE International Conference on*, IEEE, (2012).
8. Wang, Kai, et al., ***Detecting and mitigating interest flooding attacks in content-centric network***, *Security and Communication Networks* 7.4 (2014): 685-699.
9. Choi, Seungoh, et al., ***Threat of DoS by interest flooding attack in content-centric networking***, *Information Networking (ICOIN), 2013 International Conference on*, IEEE, (2013).
10. Lauinger, Tobias, et al., ***Privacy implications of ubiquitous caching in named data networking architectures***, *Technical Report TR-iSecLab-0812-001, ISecLab, Tech. Rep.* (2012).

 ISSN 2572-4975 (Print), 2572-4991 (Online)

11. Ion, Mihaela, Jianqing Zhang, Eve M. Schooler. ***Toward content-centric privacy in ICN: Attribute-based encryption and routing***, *Proceedings of the 3rd ACM SIGCOMM workshop on Information-centric networking*, ACM, (2013).
12. Ribeiro, Igor, et al., ***On the possibility of mitigating content pollution in content-centric networking***, *Local Computer Networks (LCN), 2014 IEEE 39th Conference on*, IEEE, 2014.
13. Gasti, P., et al., ***DoS & DDoS in named-data networking***, *arXiv preprint arXiv:1208.0952* (2012).
14. Xie, Mengjun, Indra Widjaja, Haining Wang, ***Enhancing cache robustness for content-centric networking***, *INFOCOM, 2012 Proceedings IEEE*, IEEE, (2012).
15. AbdAllah, Eslam G., Mohammad Zulkernine, Hossam S. Hassanein, ***Detection and prevention of malicious requests in ICN routing and caching***, *Computer and Information Technology, Ubiquitous Computing and Communications, Dependable, Autonomic and Secure Computing, Pervasive Intelligence and Computing (CIT/IUCC/DASC/PICOM), 2015 IEEE International Conference on*, IEEE, (2015).
16. Afanasyev, Alexander, et al., ***Interest flooding attack and countermeasures in Named Data Networking***, *IFIP Networking Conference, 2013* IEEE, (2013).
17. Acs, Gergely, et al., ***Cache privacy in named-data networking, Distributed Computing Systems*** *(ICDCS)*, *2013 IEEE 33rd International Conference on*, IEEE, (2013).
18. Chaabane, Abdelberi, et al., ***Privacy in content-oriented networking: Threats and countermeasures***, *ACM SIGCOMM Computer Communication Review* 43.3 (2013): 25-33.
19. Lauinger, Tobias, et al., ***Privacy risks in named data networking: what is the cost of performance?***, *ACM SIGCOMM Computer Communication Review* 42.5 (2012): 54-57.
20. Kim, Dohyung, et al., ***Efficient content verification in named data networking***, *Proceedings of the 2nd International Conference on Information-Centric Networking*, ACM, (2015).
21. DiBenedetto, Stephanie, Christos Papadopoulos, ***Mitigating poisoned content with forwarding strategy***, *Computer Communications Workshops (INFOCOM WKSHPS), 2016 IEEE Conference on*, IEEE, (2016).
22. Baugher, Mark, et al, ***Self-verifying names for read-only named data***, *Computer Communications Workshops (INFOCOM WKSHPS), 2012 IEEE Conference on*, IEEE, (2012).