Far East Journal of Electronics and Communications
© 2017 Pushpa Publishing House, Allahabad, India
http://www.pphmj.com
http://dx.doi.org/10.17654/EC017040863
Volume 17, Number 4, 2017, Pages 863-876 **ISSN: 0973-7006**

COLLISION MITIGATION SCHEME FOR NDN-RIOT-OS BASED INTERNET OF THINGS

Illa Ul Rasool, Yousaf Bin Zikria, Heejung Yu and Sung Won Kim[*]

Department of Information and Communication Engineering
Yeungnam University
South Korea
e-mail: illaulrasool@ynu.ac.kr; yousafbinzikria@ynu.ac.kr
heejung@yu.ac.kr; swon@yu.ac.kr

Abstract

Named data networking is considered to be the future of networking. The main reason being the paradigm shift from host centric to information centric communication. As it is still in the development process, it cannot be considered for realistic implementation yet. NDN supports certain unique processes such as content caching, name based content retrieval and content-decision based forwarding. These processes are supported by the strategy layer. However, during packet forwarding, the collision of data packets at receiving-end nodes results in considerable network congestion. Our work tests the NDN on RIOT-OS powered IoTs. RIOT-OS powered IoTs create the actual NDN dynamics. We further propose a forwarding strategy based congestion control scheme at strategy layer to prevent any packet collisions in network. The results show that the proposed scheme is flexible, effective and efficient for futuristic IoTs.

Received: March 13, 2017; Accepted: April 29, 2017

Keywords and phrases: NDN, RIOT-OS, IoT, ICN, CCN, TCP/IP, CS, FIB, PIT.

A preliminary version of this paper was presented at the ICIDB-2016, Seoul, South Korea.

[*]Corresponding author

Communicated by Gyanendra Prasad Joshi

I. Introduction

The futuristic Internet is becoming more application service oriented, which further procreates a requirement for scalability, reliability and security [1, 2]. Future Internet's inclination towards application centric networking is due to the recent rise in involvement of social media based networking. Information centric networking (ICN) [3-7] is one such project that has transferred networking emphasis from host centric (IP-based) to data centric networking such as named data networks (NDNs). NDN is basically a name based networking scheme where communication is based on the data-name (name of the data) instead of end points to which data may be dissipated. NDN almost follows the same TCP/IP principles at network layer. However, instead of locations of data packets, it considers using their names. This concept of networking has the ability to be able to realize communication faster than traditional TCP/IP networks [8]. The reason being NDN's ability to meet information requirements of various applications even without considering node-identity. Moreover, NDN's architecture allows it to integrate security, network traffic regulation, and routing, and forwarding strategy management within the data packet itself. This can be also related to the nonexistence of transport layer, port and sequence numbers in NDN [9]. In NDN, the transport control is regulated by supporting libraries of applications and the strategy module of the forwarding plane [9].

NDN communication relates to the concept of publisher/subscriber methodology. Instead, NDN defines subscriber as a consumer and publisher as a provider of services or data. Consumers publish interest packets containing the information regarding the required data content/packet. In response, a provider which meets these information requirements of interest packet replies with corresponding data packet. Finally, the data packet follows the same interface path (in reverse direction) earlier paved by the Interest packet.

In perspective of named data network (NDN) [10] transport control/ forwarding strategy, serves the two fundamental packet flows: (1) interest flow from consumer to provider/s, and (2) data flow from provider/s to a consumer. Both flows have some limitations. However, our work is mainly focused on Data flow from provider/s to consumer. The main reason is congestion caused due to the absence of CSMA/CA [9]. This occurs when multiple providers respond to the same interest packet at the same time. This causes considerable amounts of delay. These delays further result in increased packet loss rate (PLR) and overall network delay, and decreased interest satisfaction rate (ISR), and throughput at consumer node. We study the working of NDN on RIOT-OS [11] powered IoT devices. This provides the real-system NDN network configuration and environment dynamics. Our tests confirm the presence of congestion caused due to the absence of channel access mechanism in NDN. Further details of experimentation are discussed in later Sections II and III. In response to this, we propose a forwarding strategy to mitigate the channel congestion. Our simulations and tests show enormous decrease in PLR, delays and an incredible increase in ISR and throughput rates.

The recent transport protocols that deal with congestion are mainly receiver-driven and perform hop-by-hop interest shaping. It means that the congestion control on the data on its way back to consumer is controlled by the flow of interests generated, however following such backpressure mechanism is further complicated by other factors such as varying packet size, traffic burst, and distinct arrival time during data reception and asymmetric link bandwidth. Hence, there is unique independence between interest and data packets during upstream and downstream flows, respectively, [9, 12]. Further hop-by-hop transport congestion control is more complicated and lesser flexible, hence such tasks must be implemented on the endpoints only [13].

In [14, 15] are the receiver-driven transport protocols which control the flow of packets by a regulation mechanism at receiver. In [14], the authors propose TCP based congestion control to the receiver-driven CCN, providing fairness among ICN flows, and among ICN and TCP packet flows. In [15], authors take the issue of content popularity into consideration, and derive a throughput gain by making performance comparisons between CCN with AIMD and TCP with AIMD.

This paper presents the real testbed analytical results of NDN on RIOT-OS powered devices in Section III. It also discusses the packet collision problem of basic NDN when considering multiple providers and a single consumer scenario. Later in Section III, a controlled data flow scheme is proposed to mitigate the packet collisions. Moreover, graphical representations of results testify the efficiency and effectiveness of proposed scheme.

II. Named Data Networking

NDN is one of the main projects of ICN for futuristic Internet technologies such as IoTs sponsored by national science foundation. Other than this ICN is also involved in projects like data-oriented network architecture (DONA) [16], content centric network (CCN) [17], publish subscribe Internet technology (PURSUIT) [18], and scalable and adaptive Internet solutions [19], etc.

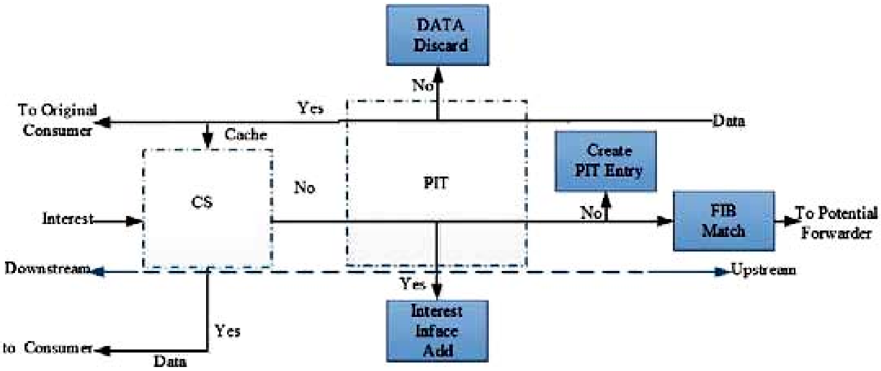

Figure 1. NDN forwarding scheme.

In future, NDN is supposedly going to play a major role in redefining the wireless networks [1, 2]. Next generation devices such as IoTs will require simple and effective wireless networking architecture to cope up with unwanted delays and low throughput rates exhibited by current TCP/IP networks [20, 21]. This drives NDN towards one of the most researched and invested projects of ICN for futuristic IoTs [3-7]. Further, a detailed NDN forwarding strategy and interest-data transport mechanism is represented in Figure 1.

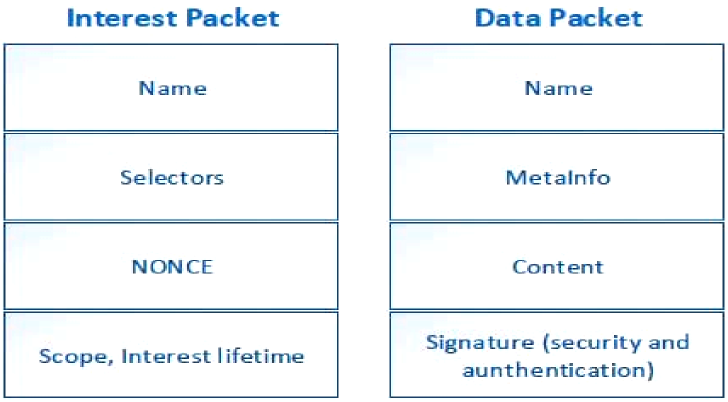

Figure 2. NDN packet structure.

NDN architecture basically constitutes of two packet types: (1) interest, and (2) data packets as described in Figure 2. Interest and data packets are defined by their respective hierarchical names. These interest/data packets are accessible by a unique name, however, it is not necessary that name must signify the function of packet. NDN nodes constitute of three basic components: pending interest table (PIT), content store (CS) and forward information base (FIB). The PIT is responsible for storing information about that Interest/s earlier forwarded by the intermediate/router node and is/are currently unsatisfied. Every data packet that traverses across intermediate nodes/routers is cached in their content store (CS) to serve subsequent requests. Content caching increases the network effectiveness in terms of lesser delays and congestion [22]. Finally, the information contemplated by FIB is utilized by intermediate routers to forward the pending interest to a potential data-provider. FIB has many name-prefix based routing protocol and constitutes of multiple output interfaces for each prefix.

When a consumer node is in requirement of any data, it dissipates an interest packet with data requirement information (data name, interest lifetime, NONCE value, etc.). Any nearby provider/s that receive an interest packet, process the following upstream operations:

- On receiving an interest packet, the first step the node performs is a content store lookup. Content store look up checks if the node has a corresponding data for incoming Interest packet. If the required data is present in CS, then the node (provider here) returns the data packet on the same interface from which Interest arrived. Otherwise the node checks its PIT entries, if a PIT match for incoming interest exists; it records the incoming interface of this interest in the PIT entry. Contrarily, in the absence of a matching PIT entry, the node (router here) creates a new PIT entry and forwards the interest towards potential data producer/s. The forwarding is done based on the information in FIB. Algorithm 1 provides detailed information about interest reception process.

Algorithm 1: Received Interest in basic NDN

```
Received Interest [Name, Selector(s), NONCE]
if Content Not in CS then
    if Name in PIT then
            Drop Interest.
    else    // if name is not in PIT.
            Add [Name, NONCE, Face] in PIT.
            Initialize Timer(s).
            Forward Interest using FIB.
    endif
else
    DATA[Name, MetaInfo, Content,]
    Send DATA.
end if
```

Moreover, when a producer node receives the interest packet, it dissipates the data back to the original consumer. The data packet follows the same route as interest packet did. When a node receives data packet, it performs the following downstream mechanism:

- On receiving the data packet, the node performs the PIT match. In case of PIT match, node checks whether data name satisfies its interest packet sent earlier. In that manner, node (consumer here) acknowledges the data through a signature verification process. Otherwise forwards the packet downstream towards original consumer. In case of unmatched PIT entry, node simply discards the data packet. The data reception mechanism is represented in Algorithm 2.

Algorithm 2: Received DATA in Naïve VNDN

```
Received [Name, Content]
if Name in PIT then
    if Face not in Application then
        Forward DATA to FIB.
        Remove [Name, NONCE, Face] from PIT.
    else
    Node Received DATA.
    end if
else
    Drop DATA.
end if
```

Nonetheless congestion occurs when long delays or lost packets occur due to buffer overflow at nodes [3, 9]. One such congestion occurs at consumer node during reception of data packets. This is mainly caused due to exaggerated flow of interest/data packets that the network can actually handle. Therefore, transport control mechanism must be designed for end users to control the flow of packets to avoid packet buffer overflow at bottleneck. Additionally, traditional implicit congestion detection techniques are not always applicable in NDN. These are mainly due to timeout feature and duplicate acknowledgments, which occur during a packet loss event.

III. Named Data Network Propagation in RIOT-OS: Testbed and Analysis

In [9], NDN is extensively studied using RIOT-OS based IoTs. This motivated us to perform real time testbed evaluation on NDN-RIOT-OS based IoT devices. Beginning with, we considered a real testbed scenario of multiple providers and a single consumer. The detailed simulation parameters follow the real time scenario constituting 1-4 providers, shown in Figure 3.

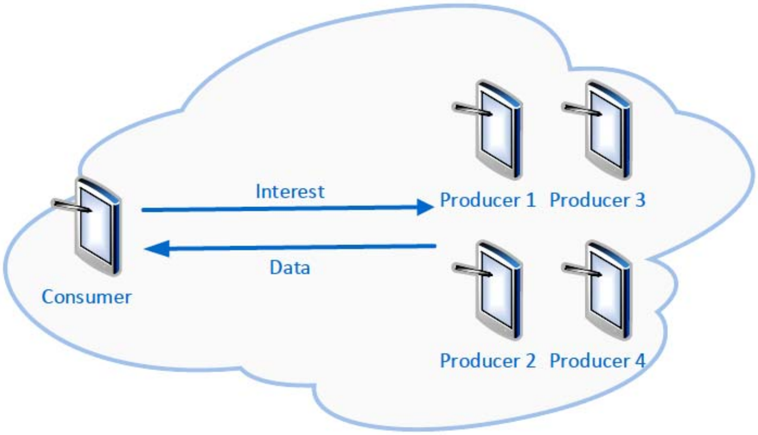

Figure 3. Scenario(s) of single consumer and (1-4) producers.

We study the working of IoT devices in correspondence to given real system scenarios 1-4. Firstly, we consider one consumer and one producer IoT devices working on RIOT-OS in base NDN networking scheme. The results show no amount of congestion or packet losses because of absence of any other provider sending similar data packets for same interest. However, as the number of producers for a single NDN-consumer is increased; the amount of packet collision also increases. This, in return, results in higher packet loss rates (PLR), and decreased number of interests that are satisfied (ISR), respectively. Concluding to this, one of the efficient ways to implement such congestion control is on end nodes (consumer or provider)

[11]. In our work, we lay emphasis on the controlled data flow from multiple providers to a single consumer. So, basically our work is based on preventing the data packet collision at consumer node.

IV. Controlled Data Packet Flow Scheme

It is presumed that every provider in the multiple provider scenarios has the required data packet that satisfies the same interest at the same time. Also, the efficient range of the devices is around 100m. This plays an effective role in NDN-RIOT-OS configuration. Every consumer node has interest timeout timer, beyond which the interest packet expires and is dropped. In NDN, it is calculated to be around 1 second or 1000000 microseconds. After 1000000 microseconds, if a consumer does not receive data packet, then it drops the Interest and fails to satisfy its requirement. Our tests show that within the range of 100m, the average round trip time (RTT) of NDN packet from consumer to producer is 845721 microseconds. So basically this gives every consumer enough time to satisfy the interest. However, due to packet collision at consumer node, the timer expires and NDN network suffers from considerable packet losses, and minimal ISR. So we propose a scheme to control the flow of Data from provider to consumer.

Considering the average RTT of 845721ms which is within the delay bound Interest timeout of 1000000ms, the remaining 154279 microseconds can still be utilized to form a random backoff window. Basically when every provider in multiple provider scenarios is sure of data availability in its CS, all of them dissipate the data packets at same time. This further as discussed leads to congestion. However, we propose that every provider after CS lookup runs a random backoff (0-130000 microseconds). This backoff range is calculated to be efficient without causing any delays as it is within interest timeout range. The 130000 microsecond is a minuscule time entity in real system scenario. Random backoff ensures that multiple providers do not overlap data dissipation times. Further, when consumer receives the required data from the smallest backoff-node, it validates and authenticates the packet received. Moreover, the consumer discards the data packets received after this because its interest is already met. This method efficiently improves the congestion rate without even modifying the event-driven interest rate generation. Further details of the Proposed Model are given in Algorithm 3.

Algorithm 3: Received Interest in Proposed NDN Scheme

```
Received Interest [Name, Selector(s), NONCE]
if Content Not in CS then
        if Name in PIT then
                Drop Interest.
        Else  // if name is not in PIT.
                Add [Name, NONCE, Face] in PIT.
                Initialize Timer(s).
                Forward Interest using FIB.
        endif
else
        DATA[Name, MetaInfo, Content,]
        Random backoff [0-130000 ms] //within the delay bound.
        Send DATA.
end if
```

All the results are averaged over five testbed runs. In Figures 4 and 5, as the number of providers increases with respect to a single consumer node, proposed scheme shows considerable improvement in interest satisfaction and packet loss rates over the basic NDN model. The packet losses for basic NDN are subjected to the cause of increasing number of producer nodes. Increasing producer number is not exclusively responsible for such losses but the time instant they dissipate the data back to producer. Elaborating the problem further, the time instants for all nodes responding with data is same. As there is no CSMA guidance, the nodes are not aware of each other’s status. This eventually leads the congestion in the network. However, when considering our proposed scheme, each node after checking the CS for available data initiates a random backoff process. After initiating the random backoff process, each node dissipates the data randomly. The consumer node receives the data from the consumer with earliest backoff time. The significance of the proposed scheme is that though as the number of producers is increasing, the scale of contention window is very large for real system devices and environment considering the 100m range. The distinct random times result in drastic improvements in PLR rates by 70% from basic NDN in single consumer-multiple (four) producer scenario. As well the interest satisfaction rate also improves drastically by same amount from the basic NDN. One of the main reasons for such a drastic improvement is, consumer node receives first data packet and satisfies its interest. Moreover, when other data packets arrive to satisfy an already satisfied interest, they get dropped immediately at verification. Therefore, the proposed scheme is effectively able to eliminate any case leading to packet collision.

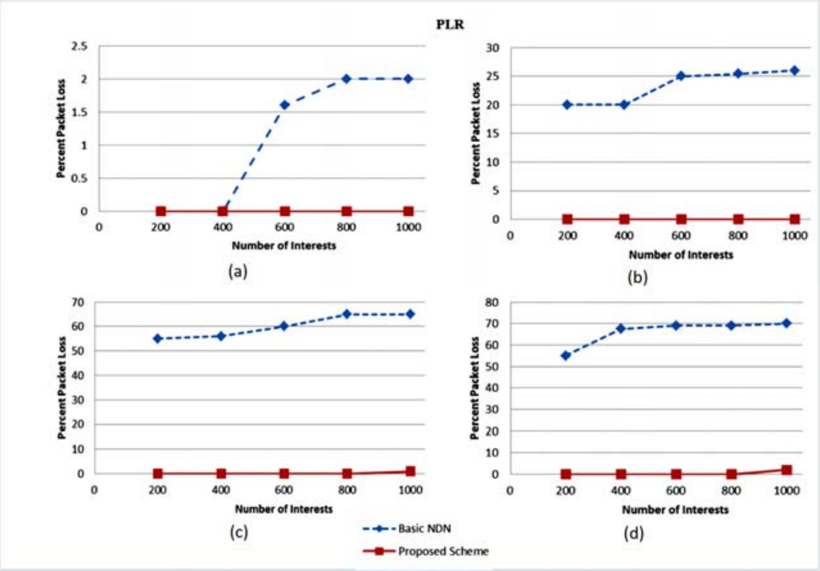

Figure 4. Packet loss rate comparison between basic NDN and proposed scheme for (a) one consumer and one producer, (b) one consumer and two producers, (c) one consumer and three producers, and (d) one consumer and four producers.

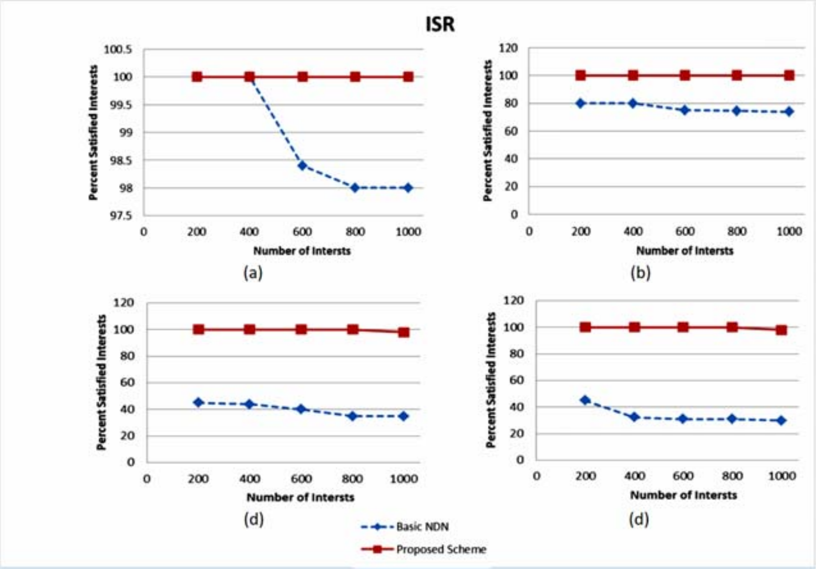

Figure 5. Interest satisfaction rate comparison between basic NDN and proposed scheme for (a) one consumer and one producer, (b) one consumer and two producers, (c) one consumer and three producers, and (d) one consumer and four producers.

V. Conclusion

We studied the basic networking technique of named data networking on NDN-RIOT-OS powered IoT devices. It is inferenced that NDN is a network technology suitable for future. However, some improvements are required to be made in forwarding mechanism to prevent the problems of congestion. We proposed an efficient and convenient random backoff technique to prevent the packet collisions. Our technique eliminated the packet losses occurred and showed significant improvement in PLR and ISR. In future research, we are planning to optimize the packet priority in NDN on real systems.

Acknowledgement

This work was supported by the 2017 Yeungnam University Research Grant.

References

[1] Rana Asif Rehman, Jong Kim and Byung-Seo Kim, NDN-CRAHNs: named data networking for cognitive radio ad hoc networks, Mobile Information Systems 2015 (2015), Article ID 281893, 12 pp. http://dx.doi.org/10.1155/2015/281893.

[2] B. Haibo, L. Sohraby and C. Wang, Future Internet services and applications, IEEE Netw. 24(4) (2010), 4-5.

[3] G. Xylomenos, C. Ververidis and V. Siris, A survey of information-centric networking research, IEEE Commun. Surv. Tut. 16(2) (2014), 1024-1049.

[4] B. Ahlgren, C. Dannewitz and C. Imbrenda, A survey of information-centric networking, IEEE Commun. Mag. 50(7) (2012), 26-36.

[5] C. Liang, F. R. Yu and X. Zhang, Information-centric network function virtualization over 5G mobile wireless networks, IEEE Netw. 29(3) (2015), 68-74.

[6] C. Fang, F. R. Yu, T. Huang, J. Liu and Y. Liu, A survey of green information-centric networking: research issues and challenges, IEEE Commun. Surv. Tuts. 17(3) (2015), 1455-1472.

[7] Chengchao Liang and F. Richard Yu, Virtual resource allocation in information-centric wireless virtual networks, 2015 IEEE International Conference on Communications (ICC), IEEE, 2015. DOI: 10.1109/ICC.2015.7248935.

[8] L. Zhang, A. Afanasyev and J. Burke, Named data networking, ACM SIGCOMM Comput. Commun. 44(3) (2014), 66-73.

[9] Q. Chen, R. Xie, F. R. Yu, J. Liu, T. Huang and Y. Liu, Transport control strategies in named data networking: a survey, IEEE Commun. Surv. Tut. 18(3) (2016), 2052-2083. DOI: 10.1109/COMST.2016.2528164.

[10] NDN Project [Online]. Available: http://named-data.net/.

[11] I. Rasool, Y. Zikria, A. Musaddiq and S. W. Kim, RIOT-OS: operating system for future IoTs, International Conference on Information and Communication Technology and Digital Convergence Business (ICIBD-2016), South Korea, 2016.

[12] T. Koponen, M. Chawla and B.-G. Chun, A data-oriented (and beyond) network architecture, Proc. ACM SIGCOMM Conf. Appl. Technol. Architect. Protocols Comput. Commun. Kyoto, Japan, August 2007, pp. 181-192.

[13] V. Jacobson, D. K. Smetters and J. D. Thornton, Networking named content, Proc. 5th Int. Conf. Emerg. Netw. Exp. Technol. (CoNEXT'09), Rome, Italy, December 2009, pp. 1-12.

[14] FP7 PURSUIT Project [Online]. Available: http://www.fp7-pursuit.eu/PursuitWeb/.

[15] FP7 SAIL Project [Online]. Available: http://www.sail-project.eu/.

[16] G. Zhang, Y. Li and T. Lin, Caching in information centric networking: a survey, Comput. Netw. 57(16) (2013), 3128-3141.

[17] Z. Ming, M. Xu and D. Wang, Age-based cooperative caching in information-centric networks, Proc. INFOCOM Workshops, Orlando, FL, USA, March 2012, pp. 1-8.

[18] I. Psaras, W. K. Chai and G. Pavlou, Probabilistic in network caching for information-centric networks, Proc. ACM Workshop Inf.-Centric Netw. (ICN'12), Helsinki, Finland, August 2012, pp. 55-60.

[19] S. Eum, K. Nakauchi, Y. Shoji, N. Nishinaga and M. Murata, CATT: cache aware target identification for ICN, IEEE Commun. Mag. 50(12) (2012), 60-67.

[20] J. Rexford and C. Dovrolis, Future Internet architecture: clean-slate versus evolutionary research, Commun. ACM 53(9) (2010), 36-40.

[21] A. Feldmann, Internet clean-slate design: what and why? ACM SIGCOMM Comput. Commun. Rev. 37(3) (2007), 59-64.

[22] Gergely Acs, M. Conti, P. Gasti, C. Ghali and G. Tsudik, Cache privacy in named-data networking, 2013 IEEE 33rd International Conference on Distributed Computing Systems (ICDCS), IEEE, 2013.