Named Data Networking: Nuts and Bolts

Subjects: Others

Contributor: Ahmed Benmoussa , Chaker Abdelaziz Kerrache , Carlos T. Calafate , Nasreddine Lagraa

Named Data Networking (NDN) is an implementation of Information-Centric Networking (ICN) that has emerged as a promising candidate for the Future Internet Architecture (FIA). In contrast to traditional networking protocols, NDN's focus is on content, rather than the source of the content.

blockchain data authentication Named Data Networking (NDN)

1. Introduction

Vehicular networks encounter many challenges, such as high mobility, heterogeneous communication, and low latency, which need to be addressed to ensure long-distance and reliable connections. However, the TCP/IP model that the actual Internet relies on was not designed for such networks as it does not adequately address these prerequisites. Furthermore, the current Internet architecture was not designed to handle the massive number of connected devices, which is expected to reach almost 30 billion in 2023 [1]. Additionally, people's use of the Internet has changed, with their focus shifting from the origin of the content to the content they want to access. To tackle these limitations, the research community has explored alternatives to support these requirements. One such alternative is Information-Centric Networking (ICN) [2], which is considered one of the most-attractive Future Internet Architectures (FIAs) [3]. Among the different ICN architectures proposed, such as DONA [4] and PUSUIT [5], Named Data Networking (NDN) [6] seems to be the most-promising contender for the upcoming Internet architecture. NDN finds its roots in the Content-Centric Networking (CCN) project [7], which emphasizes a content-oriented communication approach and prioritizes the data over their owner. Instead of relying on IP addresses, NDN uses data names for packet forwarding. In addition, NDN offers in-network caching capabilities, built-in multicast forwarding, mobility support, and security mechanisms. Unlike traditional networking protocols, NDN concentrates on securing content rather than communication channels.

Data signatures are required in NDN, which enables users to access and retrieve any content as long as its signature can be verified, regardless of its origin. NDN employs a mechanism that involves signing content by its producer and the authority who provided the certificate to the producer. To validate content, a user checks the authenticity of all the certificates involved within the certificate chain till it encounters a trusted or a self-signed network entity (usually a trusted authority). This process helps to enhance the trustworthiness of the data by verifying their authenticity. However, vehicular networks are prone to issues related to their highly dynamic nature. The data verification process employed by NDN may result in significant delays, which is unsuitable for sensitive applications, such as sharing safety messages. In traditional centralized authentication systems, data authentication is often associated with high overheads due to the need to communicate with a central authority.

2. Named Data Networking: Nuts and Bolts

NDN is an Internet architecture that prioritizes data and intends to substitute the present TCP/IP-centric architecture of the Internet. In particular, it attempts to address the escalating need for communication that is centered on content [6]. NDN identifies two entities: *producers*, who are responsible for creating and offering content, and *consumers*, who seek and retrieve data. In order to obtain content, consumers transmit the name of the desired data via a specific packet. NDN relies on a hierarchical namespace to differentiate content, for example the present paper could be named: "`com/mdpi/future-Internet/2023/ndn-bda`".

NDN relies on two packet types: *Interest* packets and *Data* packets. Consumers use *Interest* packets to request content that producers include within *Data* packets. The most-recent NDN packet specifications [8] mandate that every *Interest* packet must comprise a combination of mandatory and optional parameters. A `Name` must be included in each *Interest* packet, representing the content being sought by the consumer. The `Nonce` field, consisting of a randomly generated string of four octets, is leveraged to uniquely identify *Interest* packets, and thus prevent looping in the network. *Interest* packets may include optional parameters: `CanBePrefix` for the *Interest* packet's name, `MustBeFresh` for the requested content, `InterestLifeTime` representing how long an NDN router will maintain the state for this *Interest*, being `HopLimit` and `ForwardingHint` utilized in forwarding. In addition, *Interest* packets may also include application-specific parameters in the `ApplicationParameters` field. Furthermore, *Interest* packets can be signed if required [9].

Once a consumer sends an *Interest* packet to request particular data, the corresponding response is transmitted through a *Data* packet. In NDN, the producer should sign every *Data* packet. Fundamentally, a *Data* packet comprises three components: the `Name`, the `Content`, which denotes the payload of the *Data* packet, and a `Signature`. Supplementary details such as `ContentType`, `FreshnessPeriod`, and `FinalBlockId` are also encompassed within the *Data* packet.

Each node with the NDN stack maintains three data structures [10]. The Pending Interest Table (PIT) stores all the not-yet-satisfied *Interest* packets, which remain in the PIT until a *Data* packet returns or times out. The PIT entry contains fields such as the name of the requested *Data*, incoming and outgoing interfaces, and an expiry timer. The Content Store (CS) enables NDN to offer in-network caching, where each NDN node can store passing *Data* packets in a local cache. To maintain its size, each CS needs to implement a caching policy, such as First In First Out (FIFO), Least Recently Used (LRU), and Least Frequently Used (LFU) [11]. The Forwarding Information Base (FIB) is used to forward incoming *Interest* packets to upstream nodes. FIB entries are indexed with name prefixes instead of IP addresses, and each FIB entry consists of a name prefix and a list of next hops. Routers can forward *Interest* packets to one or multiple hops, enabling multi-path forwarding based on their strategy.

2.1. Security Principles in NDN

In a named data network, the primary objective is to locate and retrieve data objects within a specified context by their names. The security features of the NDN design are intrinsic and require signatures to be used in *Data*

packets [12]. Producers are responsible for digitally signing all *Data* they create, which enables consumers to validate the authenticity of the received information. Additionally, routers and repositories can store *Data*, and consumers can receive *Data* packets from any source. The NDN architecture relies on several security components to ensure data security [13].

2.1.1. Digital Signatures

Digital signatures in NDN represent the basic building blocks of object security. They enable securing data regardless of their location or transmission channel. The signature field is present in every *Data* packet and serves to bind the content of the packet to its name [14]. It consists of two components, namely `SignatureInfo` and `SignatureValue`. The former contains the producer's public key name and the cryptographic algorithm used to sign the *Data*, while the latter represents the signature's generated bits. The `SignatureInfo` field may also include `KeyLocator`, which includes information regarding the location of parent certificate(s). The signed hash of a *Data* packet covers all its fields except the `SignatureValue`, as illustrated in **Figure 1**.

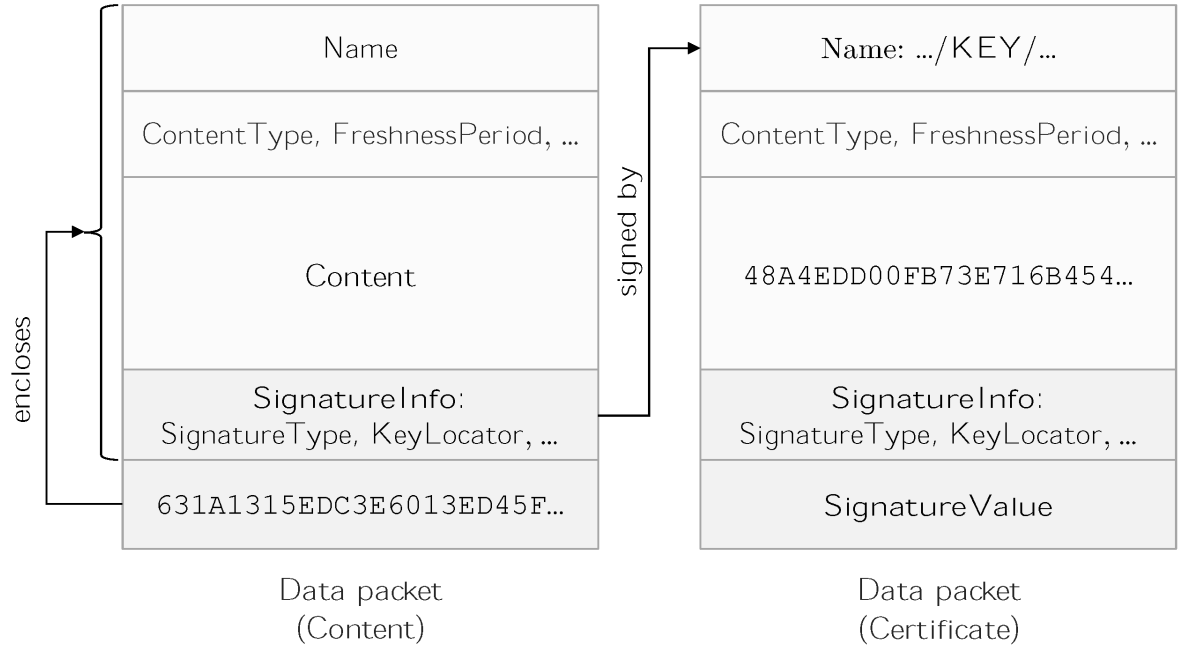

Figure 1. Example of a *Data* packet and its signing certificate.

While NDN does not mandate the use of signatures in *Interest* packets, they may be required in some scenarios where their authenticity is necessary, such as sending a new route announcement or a command packet to an IoT device. Compared to the signature field in *Data* packets, the *Interest* packet's signature field includes additional components, such as `SignatureNonce`, `SignatureTime`, and `SignatureSeqNum`, to add uniqueness to the signature.

2.1.2. NDN Certificates

To ensure secure communication in NDN, data producers need to possess at least one cryptographic key pair. Producers use their private keys to sign *Data* packets, and consumers verify them using public keys. Each public key is encoded in an X509 format digital certificate signed by an issuer [15]. The NDN certificate's name follows a specific naming convention: `/<IdentityName>/KEY/<KeyId>/<IssuerId>/<Version>`. `IdentityName` represents the namespace in which the key can operate, followed by the keyword KEY. The `KeyId` identifies an instance of the public key. Its value can be an 8-byte-long random value, an SHA-256 digest of the public key, a timestamp, or a numerical identifier. The `IssuerId` part identifies the issuer of the certificate. Its value is identical to the `KeyId`. Finally, the version of the certificate is included. In addition to its name, a certificate *Data* packet contains other fields: the `FreshnessPeriod` and the `Content`, which embeds the bits of the DER-encoded public key. Like any NDN Data packet, certificates contain a signature field, which is the signature of the issuer.

2.1.3. Trust Model

The *Data* packet's signature validates the authenticity of the received content and its origin. However, it does not indicate whether the signer has the authorization to produce the received content [16]. To verify whether a producer is authorized to generate data under a given namespace, consumers require a mechanism to check if the producer's key has the right to sign a *Data* packet under that namespace [17]. Application-based trust policies define which keys have the authorization to sign which *Data*, as illustrated in **Figure 2**. Furthermore, applications can use access control policies to protect the *Data* packet's content through encryption and permit only authorized nodes to decrypt it [18].

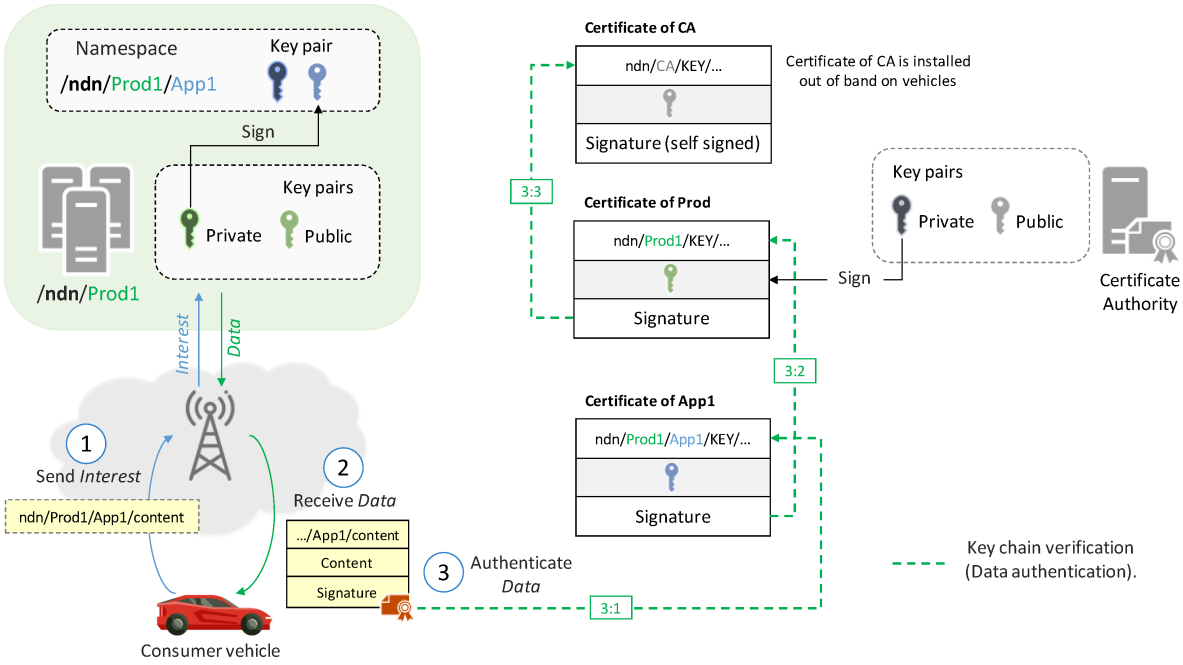

Figure 2. Example of the data authentication process.

2.2. Packet Forwarding

Instead of using IP addresses to forward packets, NDN routers utilize application namespaces [19]. NDN routers employ stateful forwarding, which implies that routers store details of received requests until they are satisfied or timed out. Upon the arrival of an *Interest* packet, the router checks whether it holds a copy of the *Data* in the CS. The NDN node forwards the *Data* downstream if it finds a copy of it. Otherwise, it checks the PIT for similar requests. If a pending request already exists, the router updates the incoming interfaces field. If no pending *Interest* exists, a new entry is created. The NDN node then forwards the *Interest* packet to the upstream neighbor(s) using a forwarding strategy and the FIB table [20]. The NDN node may respond with a NACK or drop the *Interest* packet if no route is found.

Once a *Data* packet is received by a router from a producer or in-network cache, the router checks for a matching pending *Interest* by examining the name of the received *Data* packet. If there is no match in the PIT, the router drops the received *Data* packet. However, if a match is found, the router caches (or not, depending on its caching strategy) the received *Data* and forwards them downstream to all the interfaces associated with the pending *Interest*. As a result, NDN routers provide a built-in multicast mechanism. **Figure 3** illustrates a scenario of packet forwarding in NDN.

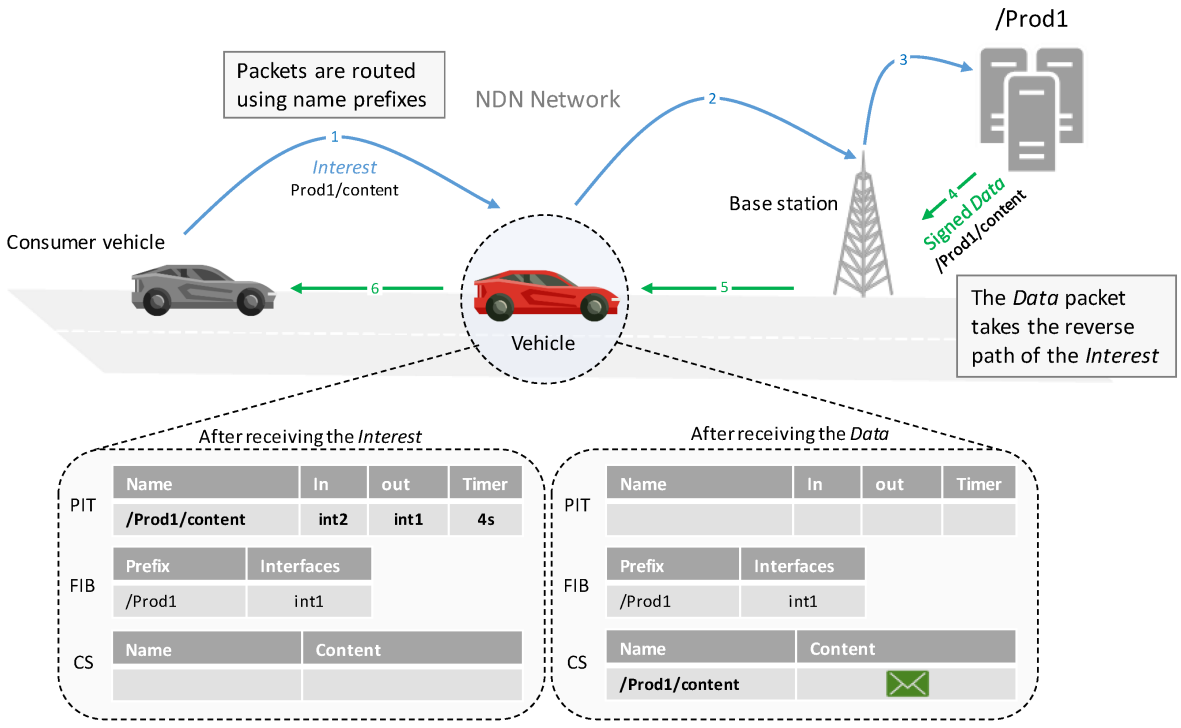

Figure 3. Example of packet forwarding in vehicular NDN.

2.3. Mobility Support in NDN

The packet forwarding process in mobile networks can be challenging due to the constantly changing network topology [21]. However, the NDN paradigm overcomes this issue by breaking the point-to-point nature of IP networks and enabling ubiquitous data retrieval. Unlike IP networks, NDN allows mobile nodes to retrieve data from anywhere in the network, instead of delivering data to a specific mobile node [22]. Furthermore, NDN's

implementation allows it to operate over any transport protocol, which is a significant advantage. Additionally, the location-independent data retrieval enabled by NDN's data-centric security is another key advantage to mobility support.

In NDN, mobile consumers are not identified by addresses, which avoids the overhead associated with address management and connection sessions that TCP/IP requires. Instead, when a consumer requests content, the *Interest* packet creates reverse path entries in the PIT as it travels to the data producer. This enables a mobility-friendly forwarding process, as *Data* packets can follow the reverse path to reach the consumer. If the consumer moves to a new location, resending the *Interest* packet is sufficient to update the reverse route to the new location [23].

The simplest way of retrieving the content of a mobile producer consists of broadcasting *Interest* packets throughout the network until they reach the intended mobile producer or the network cache holding the content. However, this approach consumes significant network resources, making it unsuitable for large networks. Another approach involves using a DNS-like node to map produced data names to the actual location of a mobile producer [24]. A consumer sends a query to the *rendezvous* point to ask for the position of the producer. The *rendezvous* point can also act as a relay, which forwards requests to the mobile producer's position and sends back data to the requesting consumer. Data depots are another approach that leverages the built-in caching capabilities of NDN, allowing consumers to send *Interest* packets to a data depot instead of directly retrieving content from mobile producers [25]. The data depot can then send the requested data if it holds them or fetch them using one of the other approaches. To retrieve data associated with a specific region, a geographic-based forwarding strategy is used where any mobile node in that region can act as a producer. For example, a consumer can request traffic information for a particular road by sending an *Interest* packet to the relevant region using geographic-based routing. When a node in that region receives the request, it either sends the requested data (if it already has them in its cache) or generates them before sending them back to the consumer.

References

1. Cisco Annual Internet Report (2018–2023) White Paper. Available online: https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-Internet-report/white-paper-c11-741490.html (accessed on 7 March 2023).
2. Xylomenos, G.; Ververidis, C.N.; Siris, V.A.; Fotiou, N.; Tsilopoulos, C.; Vasilakos, X.; Katsaros, K.V.; Polyzos, G.C. A survey of information-centric networking research. IEEE Commun. Surv. Tutor. 2013, 16, 1024–1049.
3. Pan, J.; Paul, S.; Jain, R. A survey of the research on future Internet architectures. IEEE Commun. Mag. 2011, 49, 26–36.

4. Koponen, T.; Chawla, M.; Chun, B.G.; Ermolinskiy, A.; Kim, K.H.; Shenker, S.; Stoica, I. A data-oriented (and beyond) network architecture. In Proceedings of the 2007 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, Kyoto, Japan, 27–31 August 2007; pp. 181–192.
5. Fotiou, N.; Nikander, P.; Trossen, D.; Polyzos, G.C. Developing information networking further: From PSIRP to PURSUIT. In Proceedings of the Broadband Communications, Networks, and Systems, 7th International ICST Conference, BROADNETS 2010, Athens, Greece, 25–27 October 2010; Revised Selected Papers 7. Springer: Berlin/Heidelberg, Germany, 2012; pp. 1–13.
6. Zhang, L.; Afanasyev, A.; Burke, J.; Jacobson, V.; Claffy, K.C.; Crowley, P.; Papadopoulos, C.; Wang, L.; Zhang, B. Named data networking. ACM SIGCOMM Comput. Commun. Rev. 2014, 44, 66–73.
7. Jacobson, V.; Mosko, M.; Smetters, D.; Garcia-Luna-Aceves, J. Content-Centric Networking; Whitepaper; Palo Alto Research Center: Palo Alto, CA, USA, 2007; pp. 2–4.
8. NDN Packet Format Specification Version 0.3. Available online: https://named-data.net/doc/NDN-packet-spec/current (accessed on 3 March 2023).
9. Signed Interest. NDN Packet Format Specification Version 0.3. Available online: https://named-data.net/doc/NDN-packet-spec/current/signed-interest.html (accessed on 3 March 2023).
10. Zhang, H.; Li, Y.; Zhang, Z.; Afanasyev, A.; Zhang, L. Ndn host model. ACM SIGCOMM Comput. Commun. Rev. 2018, 48, 35–41.
11. Zhang, M.; Luo, H.; Zhang, H. A survey of caching mechanisms in information-centric networking. IEEE Commun. Surv. Tutor. 2015, 17, 1473–1499.
12. Zhang, Z.; Yu, Y.; Zhang, H.; Newberry, E.; Mastorakis, S.; Li, Y.; Afanasyev, A.; Zhang, L. An overview of security support in named data networking. IEEE Commun. Mag. 2018, 56, 62–68.
13. Tehrani, P.F.; Osterweil, E.; Schmidt, T.C.; Wählisch, M. SoK: Public key and namespace management in NDN. In Proceedings of the 9th ACM Conference on Information-Centric Networking, Osaka, Japan, 19–21 September 2022; pp. 67–79.
14. Li, Y.; Zhang, Z.; Wang, X.; Lu, E.; Zhang, D.; Zhang, L. A secure sign-on protocol for smart homes over named data networking. IEEE Commun. Mag. 2019, 57, 62–68.
15. Zhang, Z.; Yu, Y.; Afanasyev, A.; Zhang, L. NDN Certificate Management Protocol (NDNCERT); Technical Report NDN-0050; NDN: Helensvale, Australia, 2017.
16. Zhang, Z.; Yu, Y.; Ramani, S.K.; Afanasyev, A.; Zhang, L. Nac: Automating access control via named data. In Proceedings of the MILCOM 2018—2018 IEEE Military Communications Conference (MILCOM), Los Angeles, CA, USA, 29–31 October 2018; pp. 626–633.

17. Nour, B.; Khelifi, H.; Hussain, R.; Mastorakis, S.; Moungla, H. Access control mechanisms in named data networks: A comprehensive survey. ACM Comput. Surv. 2021, 54, 61.
18. Lee, C.A.; Zhang, Z.; Tu, Y.; Afanasyev, A.; Zhang, L. Supporting virtual organizations using attribute-based encryption in named data networking. In Proceedings of the 2018 IEEE 4th International Conference on Collaboration and Internet Computing (CIC), Philadelphia, PA, USA, 18–20 October 2018; pp. 188–196.
19. Li, Z.; Xu, Y.; Zhang, B.; Yan, L.; Liu, K. Packet forwarding in named data networking requirements and survey of solutions. IEEE Commun. Surv. Tutor. 2018, 21, 1950–1987.
20. Voitalov, I.; Aldecoa, R.; Wang, L.; Krioukov, D. Geohyperbolic routing and addressing schemes. ACM SIGCOMM Comput. Commun. Rev. 2017, 47, 11–18.
21. Zhang, Y.; Afanasyev, A.; Burke, J.; Zhang, L. A survey of mobility support in named data networking. In Proceedings of the 2016 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), San Francisco, CA, USA, 10–14 April 2016; pp. 83–88.
22. Khelifi, H.; Luo, S.; Nour, B.; Moungla, H.; Faheem, Y.; Hussain, R.; Ksentini, A. Named data networking in vehicular ad hoc networks: State-of-the-art and challenges. IEEE Commun. Surv. Tutor. 2019, 22, 320–351.
23. Zhu, Z.; Afanasyev, A.; Zhang, L. A New Perspective on Mobility Support. NDN, Technical Report NDN-0013. 2013. Available online: https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=a6fc488646c27d13191691f7145745ba828cff9b (accessed on 30 March 2023).
24. Afanasyev, A.; Jiang, X.; Yu, Y.; Tan, J.; Xia, Y.; Mankin, A.; Zhang, L. NDNS: A DNS-like name service for NDN. In Proceedings of the 2017 26th International Conference on Computer Communication and Networks (ICCCN), Vancouver, BC, Canada, 31 July–3 August 2017; pp. 1–9.
25. Zhang, Y.; Xia, Z.; Mastorakis, S.; Zhang, L. Kite: Producer mobility support in named data networking. In Proceedings of the 5th ACM Conference on Information-Centric Networking, Boston, MA, USA, 21–23 September 2018; pp. 125–136.