Towards Named Data Networking Technology: Emerging Applications, Use Cases, and Challenges for Secure Data Communication

Afia Anjum[a,*], Paul Agbaje[a], Arkajyoti Mitra[a], Emmanuel Oseghale[a], Ebelechukwu Nwafor[b], Habeeb Olufowobi[a]

[a]*Department of Computer Science and Engineering, University of Texas at Arlington, TX 76013, USA*
[b]*Department of Computing Science, Villanova University, PA 19085, USA*

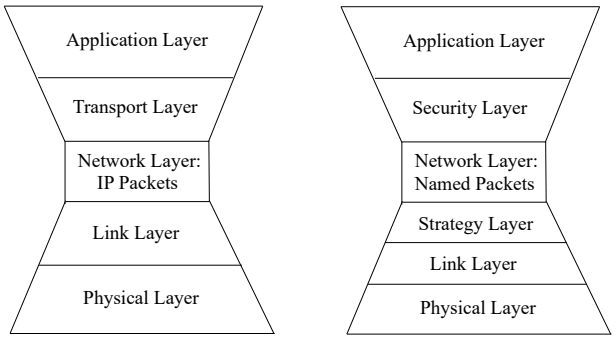
Abstract

Named Data Networking (NDN), an emerging Internet architecture is altering the basics of the networking model by making content directly addressable and routable. NDN changes the communication model by shifting from the current host-centric model to a content-centric model naming each content object hierarchically instead of using IP addresses. This change enables data consumers to request content using application-layer names that optimize network traffic and allow data security directly at the network layer, making the content of every data packet verifiable and enabling resilient communication in dynamic network environments, such as mobile ad hoc networks. This paper presents the concepts of NDN architecture and a comprehensive overview of emerging application use cases of the technology for secure data communications. We discuss the integration of NDN with the current Internet protocol and highlight how NDN works as a facilitator for addressing numerous concerns related to unique applications. Furthermore, we highlight the trust management and security aspects of NDN. Finally, we outline challenges relating to NDN adoption and present some proposed solutions.

Keywords: Named Data Networking, Trust Management, TCP/IP, Information-Centric Networking, Network Protocol, Data Communication

1. Introduction

The traditional Internet architecture, based on the Internet Protocol (IP), has provided a reliable means of data exchange between devices on the network, enabling interoperability between services and devices. The majority of today's applications' data delivery models are concerned with what data is required, regardless of its location, which is, unfortunately, not the concern of current Internet architectures. Security measures of traditional Internet architecture are channel-dependent and rely on a series of add-on features such as transport layer security [1] and datagram transport layer security protocol [2] that create overhead in establishing and maintaining secure communication channels and are unsuitable for wireless resource-constrained devices. In addition, the current IP design requires significant effort to support mobility, especially in the context of resource-constrained Internet of Things (IoT) devices due to the maximum transmission unit (MTU) mismatch [3]. Consequently, researchers sought to develop a more efficient alternative enabling content-centric communication by default where communication and data retrieval are location-independent and unique, using content names as an alternative to IP addresses to resolve these challenges. This proposal includes Information-Centric Networking (ICN), a model developed to transform Internet infrastructure to data delivery by introducing uniquely named data as a core Internet principle.



(a) IP Hourglass Architecture (b) NDN Hourglass Architecture

Figure 1: Comparison of IP and NDN hourglass architectures [4]

Named data networking (NDN) [5] is a promising candidate among several funded projects designed to enable content-based communication by default. In NDN, data is requested and retrieved using location-independent names, regardless of their hosting entity. The architecture of NDN has shifted the focus of network services from delivering packets to a specific destination address to retrieving data defined by a specific name, thus making it more suitable for today's data delivery models.

As illustrated in Fig 1, NDN maintains the IP hourglass architecture [6] with functional variations across appropriate levels. The "narrow waist" of the IP hourglass has been significantly altered by removing the constraint of designating communication endpoints using IP addresses [7]. Specifically, the hourglass architecture of NDN has enabled each content to have its

*Afia Anjum (Corresponding author) (email: afia.anjum@uta.edu).

name. The architecture has shifted the focus of network services from delivering packets to a specific destination address to retrieving data defined by a specific name [8]. The strategy layer is used as a stateful forwarding plane in NDN to make a forwarding decision for each incoming content or request, eliminating the transport layer in the protocol stack by embedding all functionalities into the forwarding plane [9].

However, despite its potential, NDN faces limitations that hinder its ability to provide a seamless communication architecture. These limitations are primarily related to caching, forwarding, and signature incorporation. In particular, caching poses storage challenges in resource-constrained mobile nodes. Additionally, content-based names introduce overhead, and signature verification adds computational costs during data retrieval. Furthermore, the hierarchical naming of content in NDN may provide some high-level information about the content's source or context, but it does not directly reveal the identity of the user who requested the content. However, patterns of content requests made for specific types of data, such as based on applications or topics, may enable attackers to infer certain information about the user's interests and preferences. To address these limitations, researchers have proposed various strategies to mitigate the challenges associated with NDN. However, previous works in this area have been limited in terms of their applications and use cases. Also, they have not comprehensively addressed the current challenges and lack exploration of an NDN trust model. As a result, there exists a gap in the literature regarding the full potential and impact of NDN.

To fully harness the potential of NDN and effectively address its challenges, researchers must gain an in-depth understanding of its underlying technology, data-centric communication model, and fundamental principles. Exploring existing solutions proposed for these challenges is crucial for comprehensively assessing NDN's current state. However, navigating the vast and increasing literature on NDN can be daunting, emphasizing the need for a comprehensive survey consolidating existing research. This paper aims to comprehensively analyze NDN as a networking protocol, evaluating its role in emerging applications and offering potential strategies to mitigate its limitations. By identifying the strengths and limitations of existing solutions, we aim to highlight areas that require further improvement and research. Additionally, we present critical research challenges and future directions to guide the advancement of NDN technology and its widespread adoption in practical applications.

Contributions. The contributions of this paper include:

- A systematic review of the literature on the adoption of NDN technology in emerging applications. This review highlights the challenges that need to be addressed to facilitate the design of efficient NDN models and promote the widespread adoption of NDN in emerging applications.

- A comprehensive trust management model for NDN technology use cases. This model serves as a foundation for identifying research gaps and provides insights for enhancing the security of NDN applications.

- An analysis of existing literature on NDN technology, which yields open research challenges. These challenges form a roadmap for future research efforts in NDN and guide the exploration of innovative solutions.

The remainder of this paper is organized as follows. We present an overview of NDN in Section 2 and review related works in Section 3. In Section 4, the usage of NDN in different application scenarios is discussed, and its security aspect is presented in Section 5. In Section 6, we discussed the challenges of using NDN and summarized the current solutions in section 7. Section 9 concludes the paper. In addition, the full forms of the acronyms used in this paper are listed in Table 1.

2. Named Data Networking (NDN): An Overview

NDN is a proposed Internet architecture that uses a content-centric communication model. Instead of addressing hosts using IP addresses, NDN names content objects hierarchically. When an application requests data, it creates an interest packet with the desired content name and sends it to the network. This naming scheme allows NDN to directly secure data at the network layer, making the content of every data packet verifiable. [10, 11].

The protocol stack of NDN inherits the hourglass model of the TCP/IP network architecture but replaces the address-based packet delivery in the "thin waist" with content names, as shown in Figure 1. In addition, the NDN stack includes two new layers: security and strategy. Applications request content using names, eliminating the need for URL-to-IP translation in the application layer. The security layer secures content using producer signatures, and the strategy layer decides on dynamic forwarding strategies based on the content's name.

The network layer provides routing functionalities using content names instead of destination IP addresses to search for the next hop using three router components: content store (CS), pending interest table (PIT), and forwarding information base (FIB). Unlike TCP/IP, where one router interface is fetched against a destination IP address, NDN FIB provides multiple interfaces against one content name [12]. The link layer and physical layer are similar to the TCP/IP link and physical layer, except the link layer maps between content names and MAC addresses instead of IP and MAC. NDN excludes the transport layer since it is a request-driven protocol where nodes control the data sending rate and request data when required, eliminating the need for a transport layer.

As mentioned above, each NDN router has three data structures: a PIT, a CS, and FIB [13]. These data structures are described in detail as follows:

1. **Pending Interest Table (PIT):** The Pending Interest Table (PIT) in the NDN architecture maintains a record of forwarded *Interest* packets that are awaiting a corresponding *Data* packet. The PIT serves two essential functions: interest aggregation and multicast forwarding. Whenever a router receives an incoming *Interest* packet, it searches the PIT entries for a matching data name. If there is no match,

Table 1: List of Acronyms

Acronym	Full Meaning	Acronym	Full Meaning	Acronym	Full Meaning
ANDaNA	Anonymous Named Networking Application	IPoC	IP Over CCN	PNL	Parallel Name Lookup
CCN	Content-centric Networking	LPM	Longest Prefix Match	PPKD	ProducerPublicKeyDigest
CDN	Content Delivery Network	MAC	Medium Access Control	PRU	Proactive Reputation Updating
CS	Content Store	MTU	Maximum Transmission Unit	QoS	Quality of Service
ESGF	Earth System Grid Federation	NDN	Named Data Networking	RPFS	Reputation-based Probabilistic Forwarding Strategy
FIB	Forwarding Information Base	NDTP	Named-Data Transport Protocol	RSA	Rivest-Shamir-Adleman
GNN-GM	Graph Neural Network-Gain Maximization	NHealthIoT	NDN-based Smart Health IoT	S-PIT	Stable Bloom Filter based PIT
HEP	High Energy Particle Physics	NPT	Name Prefix Tree	SBF	Stable Bloom Filter
ICN	Information-centric Networking	OSPF	Open Shortest Path First	TCAM	Ternary Content-addressable Memory
IoT	Internet of Things	OSPFN	Open Shortest Path First for Named Data	TCP	Transmission Control Protocol
IoV	Internet of Vehicle	P2P	Peer-to-peer	UDP	User Datagram Protocol
IP	Internet Protocol	PIT	Pending Interest Table	WSN	Wireless Sensor Network

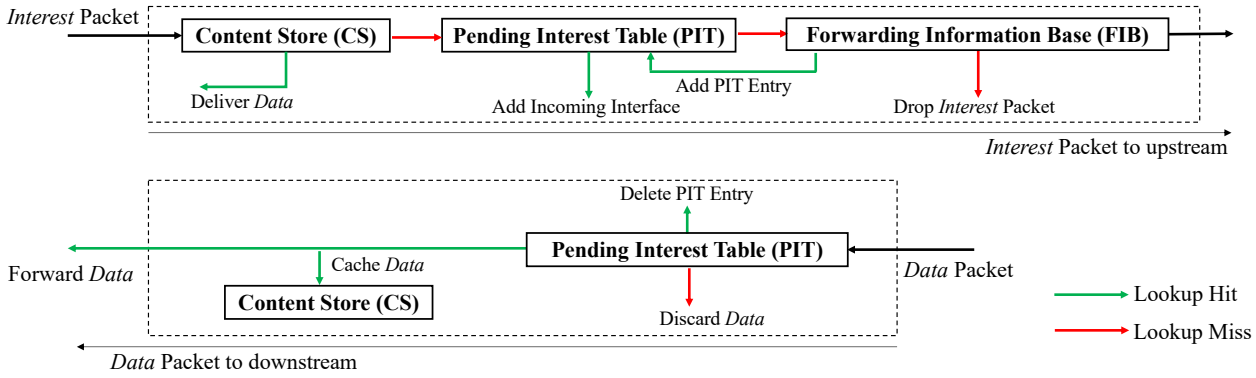


Figure 2: Forwarding Process at NDN Router Node [9]

the router creates a new entry in the PIT. If a match is found, the router appends the incoming interface number to the existing record to avoid redundant interest forwarding, a technique known as interest aggregation. Interest aggregation enables the PIT to keep track of all interfaces requesting the same data. When a *Data* packet arrives, the PIT is queried to obtain the list of interfaces that requested the data, and the *Data* packet is multicast to all of them. Moreover, the PIT also plays a role in detecting security attacks by tracking unsatisfied interests.

2. **Content Store (CS):** CS functions as a cache for each network router node, enabling NDN to achieve one of its distinctive characteristics, i.e., in-network caching. When consumers request data for the first time, the *Interest* packet is forwarded to its original producer. When forwarding the *Data* packet from the producer to the consumer, each intermittent router involved in forwarding will cache the data. If the data is requested again, any of the routers involved previously can deliver the data directly, without requiring the producer. Concretely, CS can satisfy *Interest* packets on behalf of its original producer if the data has been cached previously.
3. **Forwarding Information Base (FIB):** FIB routes *Interest* packets to the original producer or any node that has the data stored in its CS. Each FIB entry comprises the name prefix of a data item as well as a list of hops that will guide *Interest* packets to nodes requesting the data.

A representation of data exchange in NDN is illustrated in Figure 2. Here, when a user U wants to retrieve data D, it sends an *Interest* packet with the data name. The NDN router that receives the *Interest* packet checks its CS to see if it has the requested data. If it does, the router will deliver the data using *Data* packet. If there is no match for the data, the router will check its PIT to see if it has already received an *Interest* packet for the same data D. If it has, the router will add the new incoming interface ID to the existing PIT entry for that data, instead of forwarding the *Interest* again. For example, if at time $T = t_0$ user A (Interface 0) requested data D, Interface 0 will be stored in the *Interface* field of PIT entry for the *Interest* packet. Say at $T = t_1$, user U (Interface 1) requests for the same data, instead of forwarding the *Interest* again, the router will simply add Interface 1 to Interface 0 in the *Interface* field (Figure 3). This process of merging multiple *Interest* packets for the same content is called *Interest Aggregation*. When the router eventually receives the *Data* packet for this content, it will be forwarded to all interfaces that requested it, enabling multicast.

If no entry is found in PIT, the *Interest* packet is forwarded to the FIB. The FIB uses the longest prefix match (LPM) algorithm to find the appropriate next hop for the requested data. The router then creates a new entry in the PIT and forwards the *Interest* packet to the selected next hop. If there is no match in the FIB, the router will either flood the *Interest* to all outgoing interfaces or drop the *Interest* packet, depending on the forwarding strategy and protocols [9].

When NDN routers receive a *Data* packet with the data producer's signature, it checks its PIT for a matching pending *Interest* using the content name in the packet. If there is a match, the router will send the *Data* packet to all the interfaces listed in the matching entry in the PIT, and then delete the entry from the PIT. Also, the router will store the contents in its CS according to the caching policy. Based on the above example, we identify the four core elements of the NDN approach, which are as

Table 2: Summary of Previous Research

Domain	Authors	Contributions	Limitations
ICN Proposals	Xylomenos et al. [14]	Key ICN proposals highlighting the similarities and differences, and the limitations of the proposals	No description of NDN as a separate entity
	Ahlgren et al. [15]	Concepts, design, and benefits of ICN based on naming, caching, and routing of data	
	Xiaoke et al. [16]		
NDN Overview	Zhang et al. [8]	Description of NDN architecture, key components, and applications	Recent applications, research challenges classified by NDN functionalities, and the mitigation approaches not mentioned
	Saxena et al. [9]	Illustration of NDN architecture, applications, and research challenges	Current trends to address the mentioned research challenges are not discussed
NDN Functionalities	Fan et al. [17]	Analyze NDN caching approaches and provide a method to determine the caching algorithm used by network	Focused only on caching feature of NDN
	Li et al. [18]	Highlighted several lookup methods that NDN uses to search content in CS, FIB, and PIT	Focused only on the NDN forwarding plane
	Tariq et al. [19]	Describe several forwarding strategies, limitations, and solutions to improve scalable forwarding in NDN	
	Soniya et al. [20]	Survey the strategies employed to address various research challenges of the NDN forwarding plane	
	Yuan et al. [21]	Identify the challenges regarding the NDN forwarding plane and propose mitigation approaches	
	Ariefianto et al. [22]	Survey different routing techniques and highlight the research challenges	Focused only on NDN routing solutions
	Zhang et al. [23]	Survey several data mobility approaches in NDN	Focused only on NDN mobility solutions
	Shannigrahi et al. [24]	Propose integration of NDN and several scientific domains by emphasizing on the similarities in the content naming structure	Focused only on NDN naming aspects

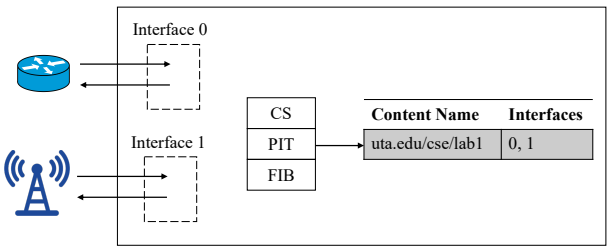


Figure 3: Interest Aggregation

follows:

- **Naming**: One of the most essential aspects of NDN design is the hierarchical naming of data, which allows each application to choose a naming structure independently from the network. For example, a video produced by ”myApp” may have the name ”/myApp/videos/demo.mpg,” where the forward slash ”/” delineates name components in text representations, similar to URLs [8]. This approach eliminates three issues that arise from using addresses in the IP architecture: address space exhaustion, NAT traversal, and address management. Moreover, NDN’s naming system increases mobility, as there is no need for mobile users to acquire IP addresses every time they switch networks [9].

- **Security**: Another key feature of NDN is its data-centric security. NDN requires data producers to sign data packets using public-key cryptography to achieve authenticity, confidentiality, and availability. NDN also automates all cryptographic key management and operations, making security highly usable for applications. Despite these benefits, NDN is not immune to attacks. In particular, interest flooding and content poisoning attacks remain significant concerns and require further investigation [25].

- **Routing and Forwarding**: In NDN, routing is responsible for setting topology and policies. NDN has three main routing protocols: OSPF-based routing [26], two-layer routing [27], and link-state routing [28]. These protocols use the FIB to store routing-related information. When an *Interest* packet is received, the router searches for the content name’s prefix in the FIB to find the next hops and forwards the packet in the correct direction. NDN allows multi-path forwarding by enabling each FIB entry to have multiple next hops [29]. The routing protocols determine the availability of routes from these multiple next-hops and select the forwarding route based on performance.

- **In-network Caching**: In NDN, data packets are identified by their names and signatures, meaning they can be cached and served anywhere in the network. This is in contrast to traditional IP networks, where data packets are identified by their source and destination addresses. While caching is not compulsory in NDN, it is a crucial feature that reduces bandwidth usage by storing copies of frequently requested data in routers and other network nodes. When a request for cached data arrives, the node can serve the data locally without having to forward the request to the original source. This can help reduce network congestion and improve performance, especially for popular content. NDN also supports a more advanced form of caching called Repository, which provides larger and more persistent storage for data that is expected to be available for longer periods of time.

3. Related Work

Content-centric networking (CCN) is a networking paradigm that aims to address the shortcomings of IP by providing refined support for client mobility, multipath connectivity, multicast delivery, and in-network caching. Over the past few years, CCN

has gained increasing attention as a new networking paradigm in various application environments [30, 9]. This section reviews related work on CCN and its proposals, focusing on the application of CCN proposals and surveyed approaches related to their implementation. A summary of these works is presented in Table 2.

Ahlgren et al. [15] and Xiaoke et al. [16] present the concepts, design, and benefits of ICN based on the components of named data object, routing, and caching. Zhang et al. [8] present a survey that introduces naming, routing, forwarding, security, and caching technologies of NDN, while Saxena et al. added the mobility aspects of NDN in their work and discussed applications in different contexts, including sensor networks, vehicular networks, and multimedia streaming [9]. Surveys specific to functionalities of NDN are presented in [19, 22, 24]. Among those, Fan et al. [17] survey different caching decision policies in NDN and developed a measuring scheme that uses the mechanism of repeatedly requesting the same content and comparing response times, which helps application developers identify which caching algorithm the network's routers are employing.

The forwarding features of NDN have been presented by [18, 19, 21, 20]. The initial requirement for forwarding content is quick lookup, which is dependent on the efficient lookup algorithm and data structures of CS, FIB, and PIT. Li et al. [18] outlined four alternative lookup methods as well as data structures employed in the NDN forwarding plane. Tariq et al. [19] explore forwarding algorithms for ad-hoc networks, such as mobile and vehicular ad-hoc networks, wireless mesh networks, and wireless sensor networks, and present different forwarding strategies, forwarding-related problems, and outline solutions to improve scalable forwarding in NDN. Similarly, Soniya et al. [20] survey the working mechanism of NDN's forwarding plane and the approaches used in mitigating specific forwarding challenges, such as prefix hijack, link failure, and congestion. In the work of Yuan et al. [21], fundamental challenges related to NDN forwarding are identified, and solutions for achieving a scalable NDN forwarding plane are proposed.

Ariefianto et al. [22] present a classification of the routing techniques used in NDN, including IP routing enhancement, geometric routing, and centralized routing, and highlight open challenges for each class. Zhang et al. [23] surveyed NDN mobility solutions, such as mapping, tracing, data depot, and data spot algorithms, and categorized them based on their design approaches. The authors also provide IP and NDN mobility support designs comparison, highlighting their features. Naming, which is a critical aspect of NDN, is surveyed by Shannigrahi et al. [24]. To facilitate the integration of NDN and scientific domains, such as climate science, high-energy particle physics, and genomics, the authors discuss the similarities of the current naming structure used in these domains with NDN's and provide a list of naming recommendations.

Despite the abundance of literature on NDN, these works have primarily focused on specific features, providing limited information on practical applications and use cases, thus leaving the full potential of NDN largely unexplored. Moreover, recent surveys on NDN have failed to adequately address the latest application challenges and mitigation approaches associated with this networking paradigm. Consequently, there exists a gap in the literature that necessitates further exploration to comprehend the true impact and potential of NDN. Therefore, this paper aims to address these gaps by presenting a comprehensive survey of key features of NDN and the challenges associated with its application. We also present several emerging use cases that demonstrate the versatility of this networking concept. Additionally, we identify and discuss the open research challenges that need to be addressed to enhance the practical implementation of NDN. Furthermore, our study introduces an NDN trust model, which has yet to be explored in the existing literature, offering a novel perspective on ensuring trust and security in NDN networks.

4. Emerging Applications of NDN and Use Cases

This section presents an overview of NDN architecture and its functional aspects in emerging applications. These applications are fundamental in realizing the smart connected ecosystem incorporating numerous communications and security technologies for controlling different assets. Moreover, we highlight the significant challenges, roles, and benefits of using the NDN model in the applications. A chart summarizing the NDN applications and use cases is shown in Figure 4.

4.1. Peer-to-Peer Data Sharing using NDN

Peer-to-peer (P2P) data sharing in Named Data Networking (NDN) is a decentralized system where each user/node functions as both a client and a server, enabling them to interact without a central server. This architecture allows for sharing of resources among equally privileged peers without the need for a central server. As more peers join the network, the sharing mechanism scales resulting in additional resources. However, when a peer leaves the network, its shared resources are also taken away. BitTorrent is an example of a P2P application that allows the distribution of large torrent files among peers in equal-sized chunks. Despite its advantages, traditional P2P network applications built on top of TCP/IP protocol face various challenges. In P2P data sharing, large files are distributed among multiple peers, but challenges arise due to the lack of a central server.

One challenge is *tracking IP addresses*, as peers may change their IP addresses when they join new networks, making it difficult to monitor them. Another challenge is the *additional overhead caused by peer selection*, as client-peers must continuously choose the most appropriate peer group to source data from. However, initially selected peers may not always provide acceptable performance due to unforeseen circumstances, such as node unavailability, which can add to latency and TCP connection overhead.

Data retrieval failure is another challenge in P2P networks since peers can leave the network after completing a download, leaving other peers unable to retrieve the data they were still downloading. Finally, *application layer overhead* is an issue

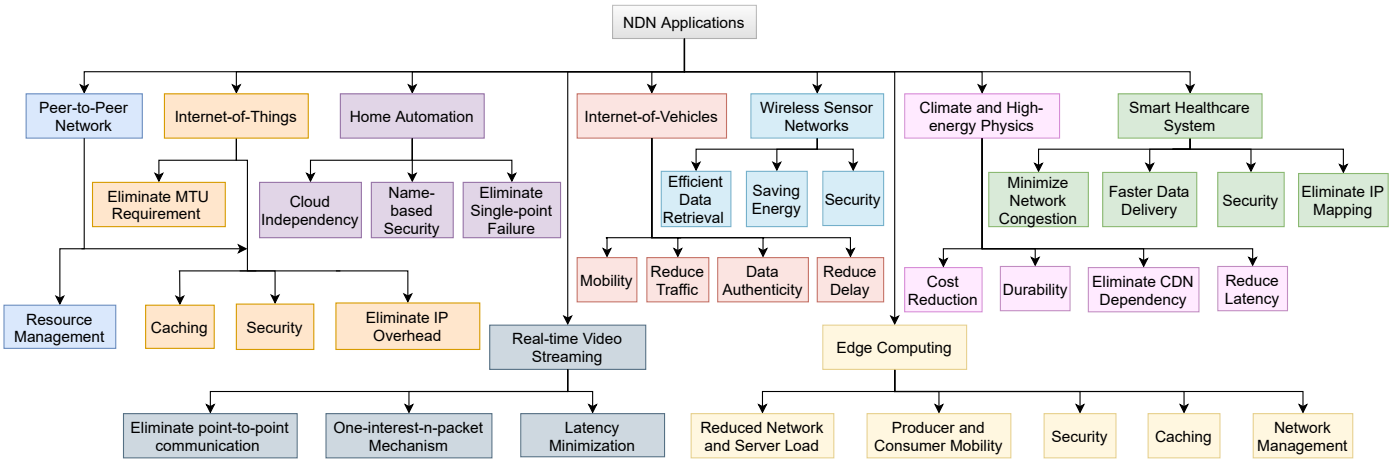


Figure 4: Emerging applications of NDN along with the use cases

since TCP/IP protocol is not designed to implement data redundancy, and data replication and security mechanisms are required at the application layer to ensure availability and protection.

The use of Named Data Networking (NDN) as a network architecture offers several advantages for peer-to-peer (P2P) data-sharing applications, as highlighted in a study by Mastorakis et al. [31]. One of the benefits of NDN is improved resource management. In traditional P2P networks, discovering the top peers for data retrieval can introduce overhead and increase latency. However, in NDN, when a peer requests data, the forwarding plane can bring the requested data to the peer through the best path in the network. This results in improved network throughput without adding additional overhead.

Another advantage of NDN is improved caching. Each NDN router acts as an in-built cache, making data replication inherent in the network layer. This caching capability improves data delivery rates, as data cached in routers in the network can quickly reach requesting peers. Furthermore, NDN caching eliminates the need for data replication in the application layer. NDN also offers enhanced security as each data fragment distributed over the network includes a digital signature from its original producer, ensuring content authenticity without the need for a secured channel in the network layer.

Lastly, NDN eliminates IP overhead in P2P data-sharing applications. Peers can retrieve individual fragments of a large file independently since each data packet has a unique name that can be used to request it. This eliminates the need to track the IP addresses of peers.

Overall, the use of NDN in P2P data-sharing applications brings significant advantages.

4.2. NDN in Internet-of-things (IoT)

The Internet of Things (IoT) is a collection of embedded devices, referred to as "things," that communicate over the internet to share data. Today, IoT protocols are deployed over the network layer using TCP/IP as the transport mechanism. However, the implementation of TCP/IP has been successful for wired network devices, such as mainframes, desktops, and laptop computers. Most IoT devices are resource-constrained and mobile, making low-energy Layer-2 technologies, such as IEEE 802.15.4, the preferred option for IoT networks. This preference is due to the small MTU size, merely 127 bytes, required by these technologies [32]. This MTU size is in contrast to the minimum MTU of 1500 bytes supported by today's IP network, which is unrealistic for constrained network devices [3].

Montenegro et al. [33] suggest an adaptation layer to solve the mismatch between the old design and new requirements. However, this layer adds additional complexity and overhead to the network. These differences pose significant challenges for IoT devices that use TCP/IP as a communication protocol. Some of the key challenges for IoT devices using TCP/IP as a communication protocol are highlighted below:

- Multicast: Multicasting is used in many IP-based network protocols to send broadcast messages to every node in a network or make a query without knowing the target node. However, this gives rise to different issues in IoT devices, such as missed packets and energy loss. Most IoT devices run on limited power, so they tend to conserve energy by going into sleep mode. This inactive mode may cause devices to miss multicasted packets. Moreover, due to multi-hop forwarding activities, devices also lose energy when they become active. In light of these issues, IP protocols need to be redesigned to satisfy some of these constraints in the IoT ecosystem.

- Routing: IoT devices are expected to roam around due to their support for mobility. However, this notion leads to overhead in keeping track of the IP addresses of IoT devices whenever they change networks.

- Caching: As there is no in-built caching policy in TCP/IP protocol, IoT applications create application-level caching policies for efficient data dissemination. The process involves choosing a proxy node that caches contents for any inactive node and forwarding data when the node wakes up. However, the dynamic nature of the network may cause the selected proxy node to become unreachable when the cached content is needed.

- Security: Applications running over TCP/IP protocol ensure security by fortifying the communication channel. This security mechanism requires that both the producer and consumer of data create and maintain a secure channel during communication. However, this approach is not suitable for IoT devices due to resource constraints. Moreover, the security of the data is not guaranteed by securing the communication channel since data contents can get modified once they are out of the secured channel.

The NDN architecture offers potential solutions to the challenges faced by IoT networks using TCP/IP and has the potential to meet the requirements of IoT communication [34]. One of the key advantages of NDN is its use of data naming conventions that eliminate the need for tracking the IP addresses of nodes, which helps reduce overhead and simplifies the routing process. Moreover, instead of multicasting packets to all nodes in the network, NDN utilizes interest packets to retrieve the desired data, which significantly reduces unnecessary traffic and energy consumption. Another benefit of NDN is that it enables in-network caching policies, which allows each hop to act as a cache, eliminating the need for proxy nodes for content caching. This feature provides efficient data dissemination and reduces latency. Finally, NDN directly secures contents, ensuring that the data communicated remains secured irrespective of the communication channel. This is achieved through the use of cryptographic signatures that guarantee the integrity and authenticity of the data, even if it is cached or forwarded by intermediate nodes. Consequently, Zhang et al. [35] proposed a unified IoT platform based on ICN architecture, where ICN represents the broad research direction under which NDN was proposed, to provide seamless mobility support, scalability, and efficient content delivery. Similarly, Askar et al. [36] incorporate NDN for IoT communication, introducing NDN-based IoT, since NDN has shown great potential with its support for mobility, caching, naming and security.

4.3. NDN in Home Automation

Home automation, or smart home systems, refers to the automated control of networked devices connected to the internet in households. These systems use sensor data to make intelligent decisions. However, they are vulnerable to cyber threats and physical intrusions. Most industrial solutions, such as Google threads, Samsung SmartThings, and ZigBee Home Automation, rely on remote cloud servers or complex network protocols that lack strong protection from unauthorized access to data.

To address these issues, researchers have proposed using Named Data Networking (NDN) to build secure home automation systems. Pi et al. [37] proposed a cloud-independent system that uses a hierarchical namespace for smart home devices. Devices joining the network must undergo trust bootstrapping to validate their identity, and their shared data is trusted within the network.

Ahmed et al. [38] proposed an NDN-based system that focuses on secure data distribution and retrieval. The system allows users to collect data from nearby sensors using the in-network caching feature of NDN. It also creates a private cloud to store data and forwards critical data to users through push-based forwarding when they are not in proximity.

Zhang et al. [39] proposed a framework called Sovereign that enables users to build self-contained smart home systems that they own and control. The framework utilizes NDN's essential features, such as its naming scheme, forwarding mechanism, and security policies. It also includes a named-based security policy that specifies who is authorized to produce and access data, and it checks the policy when authenticating a data packet.

4.4. NDN Based Real-time Video Streaming

Real-time video streaming has become increasingly popular in recent years, leading to a surge in video content. Current video delivery mechanisms are based on TCP/IP networks, where the video producer sends data directly to viewers. However, TCP/IP's point-to-point communication creates a problem when multiple clients request the same video, resulting in duplicate packets being sent. NDN's in-network caching and interest aggregation features can resolve this challenge.

Realizing the increasing demand for live video streaming and the necessity of load balancing in the network, Ramadha et al. [40] proposed a video streaming system based on NDN that significantly improves over IP-based video streaming systems. Another example of an NDN-based real-time video streaming solution is NDNVideo, proposed by Kulinski et al. [41]. NDNVideo indexes each video by time frames, allowing random access to live video streaming. A consumer issues an interest with the desired timecode and content name to retrieve the most recent data. After receiving the first data packet, the user can request video data using consecutive segment numbers. NDNVideo also ensures persistent storage of live content through caching and handles packet loss by allowing the client to skip to the most recent segment of data, enabling uninterrupted playback.

Li et al. [42] proposed an NDN-based real-time traffic support system that modifies the format of interest and data packets through the addition of a field called the type of service (TOS). TOS indicates whether the video is real-time or pre-recorded. Traditional one-interest-one-packet mechanisms are not suitable for real-time video streaming as delays are unacceptable. Additionally, real-time traffic data is only meaningful for a short time, making caching of such data useless. To speed up the process, Li et al. proposed a one-interest-n-packet mechanism where data chunks are delivered by the source as caching of real-time content is disabled. This approach reduces the delay and ensures that real-time traffic data is available as soon as it is generated.

4.5. NDN for Wireless Sensor Networks (WSN)

Wireless sensor networks (WSNs) are widely used for tasks such as logistics, environmental monitoring, and surveillance. However, WSNs face several challenges related to efficient data retrieval, energy consumption, and security. Named Data Networking (NDN), a new communication paradigm, has the potential to address these challenges and enhance the capabilities of WSNs.

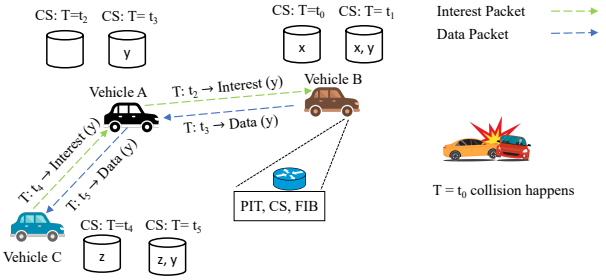


Figure 5: Risk minimization using NDN caching in IoV. At time t_0, collision happens and Vehicle *B* generates collision data *y*. Vehicle *A* initiates Interest for *y* and gets a reply from Vehicle *B*. At time t_4, Vehicle *C* forwards an Interest asking for *y* and gets a reply from Vehicle *A*, instead of the original producer *B* due to distance.

WSNs have a large number of sensor nodes, and individually addressing each node is problematic. This results in significant redundancy and consumes the node's limited battery power. NDN can help eliminate this redundancy by enabling a sensor node to request data independently using an interest packet that includes the data name.

Additionally, the energy consumed by a client node to repeatedly retrieve the same data from its original producer can be significant, especially if there are no nearby nodes having a copy of the data. The in-network caching capability in NDN helps eliminate these redundant requests by ensuring that the requested data is readily available among nearby nodes.

Furthermore, due to the low computational and storage capacity of WSNs, they cannot use expensive cryptographic techniques to enhance security. NDN's data-centric security can help ensure that secured content is transmitted and received by any node in the network, even though WSNs broadcast messages using wireless links that are susceptible to attacks. NDN's unique features, such as efficient data retrieval, energy saving, and security, can magnify the potential of WSNs and enhance their capabilities for various applications.

4.6. NDN for Internet of Vehicles (IoV)

The Internet of Vehicles (IoV) is a network of connected autonomous vehicles that share data with each other and other entities in their environment, such as roadside units, pedestrians, and buildings, as well as the cloud. This communication enables the exchange of critical information, such as road conditions, collision warnings, and traffic, using a communication protocol [43]. However, the current IP architecture used in IoV has limitations, as the dynamic nature of the entities and the constant changes in the network topology make it difficult to track IP addresses [44]. In contrast, the Named Data Networking (NDN) architecture is name-based, which eliminates the need for IP addresses, making it easier to track the nodes' mobility. Additionally, NDN's cache enables data retrieval from any nearby node that has a copy of the data, reducing data traffic. Also, security is a crucial component of IoV, as unauthorized intrusion can have catastrophic consequences. NDN inherently ensures the authenticity and integrity of data communicated in the network, making it a suitable choice for IoV.

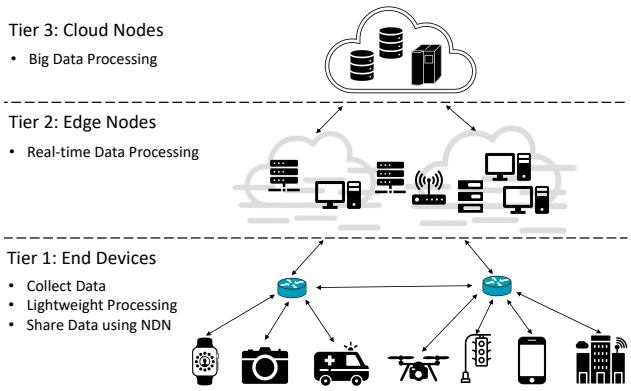


Figure 6: Fog Computing using IoV Edge Nodes

One of the essential goals of autonomous vehicles in IoV is to provide time-bound autonomous navigation to minimize the risk of accidents. NDN facilitates this through its caching and content-driven properties, which allow each vehicle to become a "data mule" that transports data opportunistically, selecting the best possible route to retrieve the data from nearby nodes and minimizing the latency of safety-critical data packets. Let us consider an NDN use case for a collision warning service using three vehicles A, B, and C, as shown in Figure 5. To get the collision information on a specific road, vehicle A will request data using an interest packet that includes the name of the desired data. Vehicle B, being near the collision zone, and equipped with different sensors and communication technologies, will detect the collision. Routing and forwarding strategies will direct the interest packet to vehicle B, and it will share the data packet with vehicle A in the reverse path by combining the data with its signature. If vehicle C, which is near vehicle B, requests the same collision data, the router will forward its interest to vehicle B instead of the original producer due to caching facilities. Upon receiving the data, vehicle C can ensure authenticity using the signature embedded in the data packet. Hence, each vehicle becomes a "data mule" that transports data opportunistically.

Various work have proposed integrating NDN architecture into IoV to address concerns regarding mobility, security, reliability, and data dissemination. For example, Barka et al. [45] propose a novel trust-aware Monitor-based communication architecture for trust-based communication between aerial vehicles, while Amadeo et al. [46] states the potential of NDN in facilitating effective communications in IoV. Chen et al. [47] explore NDN's in-network caching, and Khelifi et al. [48] propose a proactive caching-based mobility prediction strategy that uses NDN features in IoV communication to decrease content retrieval latency. Similarly, Aboud et al. [49] consider NDN as a promising solution for vehicular adhoc networks since NDN with its unique forwarding rules can optimize network resources and provide a lightweight data dissemination model.

4.7. NDN for Edge Computing

Edge computing brings computation and data storage closer to the devices that generate data. By doing this, edge computing reduces latency and improves control over cloud services.

However, edge computing using traditional TCP/IP protocols can be challenging due to issues like random disconnections, unreliability, and lack of network resiliency. To address these challenges, researchers are exploring new network protocols like NDN.

NDN offers unique routing and forwarding techniques that simplify machine-to-machine communication. By combining NDN with edge computing, researchers hope to amplify the benefits of both approaches. Specifically, authors in [50] propose a 3-tier architecture that includes NDN at Tier 1 (devices), edge computing at Tier 2 (edge nodes), and cloud computing at Tier 3 (cloud nodes), as shown in Figure 6. At Tier 1, all end users or IoT devices communicate with each other using the NDN network. This communication is content-centric, with contents being requested at the network layer using names. Tasks that cannot be handled by end devices are forwarded to edge nodes, while computations that cannot be satisfied by edge nodes are forwarded to cloud nodes for processing.

Combining NDN with edge computing has several benefits. For instance, NDN's interest aggregation feature can help control the traffic generated from end-users sent to edge devices. In this case, incoming requests for the same content will be aggregated in the PIT table and will not be forwarded further, reducing network and server loads. In-network caching can also help reduce latency by reducing the need to relay interest requests to the original producer. Finally, NDN's receiver-driven and connection-less approach helps achieve consumer mobility.

4.8. NDN in Climate and High-Energy Physics

Climate science [51], high energy particle physics (HEP) [52], and other scientific disciplines routinely create and process a massive amount of data and have relied on customized systems to manage their data.

However, using traditional systems such as a content delivery network (CDN) can be costly and pose a high risk of data loss. TCP/IP protocols, which are often used in these systems, also result in some limitations for these scientific communities. For instance, the Earth System Grid Federation (ESGF), a distributed system for managing climate data, has experienced security, latency, and repeated request issues when using TCP/IP as the communication protocol [53].

ESGF's lack of security measures, caching capabilities, and request aggregation mechanisms can be addressed by leveraging NDN's features. NDN's interest aggregation feature, for instance, can merge repeated requests for the same data, reducing the load on both the source server and the network. NDN's in-network caching feature, which secures each data packet through signature, can be incorporated to reduce network traffic, server load, and data delivery latency.

To illustrate this, consider a scenario where user A requests data D at time $t = 1s$. Assuming that this is the first time the data is requested, it will be retrieved from the producing node or server, and each node it passes through will cache the data in the CS. If user B requests the same data at time $t = 5s$, the data will be fetched from the nearest node that cached the data, instead of traveling to the producer. Thus, by leveraging NDN's features, the performance of customized systems such as ESGF can be significantly improved, reducing the complexity of developers at the application level.

4.9. NDN in Smart Healthcare System

The use of IoT-driven applications in smart healthcare systems has allowed doctors to remotely monitor, analyze, and treat patients, improving care. However, these systems rely on IP-based connectivity, which comes with certain limitations [54]. In IP-based systems, communication is established through the transport layer for end-to-end connectivity channels that facilitate data transportation and reliable delivery of packets in both wired and wireless scenarios. Unfortunately, the transport layer's inefficient design and functionality make time-sensitive data communication and retrieval difficult. The congestion control mechanism assumes that packet loss is due to congestion, which leads to a decrease in the congestion window by half. This can result in inaccurate readings and delayed emergency medical data retrieval, which could be life-threatening. Furthermore, health-related data is sensitive and needs proper security during transmission. However, IP-based networks lack built-in security, which could result in security breaches. The TCP/IP protocol suite's static node design does not work well with healthcare devices and applications' ubiquitous nature, and it is challenging to maintain IP when mobile nodes move from one network to another [55].

In contrast, NDN provides congestion control by the data consumer, which initiates and controls network transport [56]. The NDN router controls the interest forwarding rate on a hop-by-hop basis to a specific node, thus handling congestion. If congestion or delay occurs in data transmission, NDN's in-network caching feature aids delivery using the router's CS to identify where the packet got lost in the hops. By storing data near the user with in-network caching and content security, NDN reduces delays in delivering time-sensitive data while ensuring its security. Additionally, NDN eliminates the challenge of tracking IP when mobile nodes change networks using content naming, which reduces the overhead of mapping application names to IP addresses.

Saxena et al. presented work on integrating NDN in healthcare [55]. The authors developed a smart health system named NHealthIoT that uses NDN-based communication to collect sensor data. The collected data is delivered to the home server, which identifies emergency events using a hidden Markov model. The events are immediately sent to the cloud server using content-aware adaptive forwarding, while non-emergency events and health-related data are saved and can be accessed by authenticated users when required. Similarly, Saxena et al. [54] proposed a smart health-NDNoT that integrates NDN with IoT, consisting of four layers to collect, manage, store, and generate reports based on health-related data. Gupta et al. [57] proposed a secure remote healthcare monitoring framework incorporating NDN as the communication framework to exploit the advantages of NDN, such as named-based forwarding, security, and in-network caching.

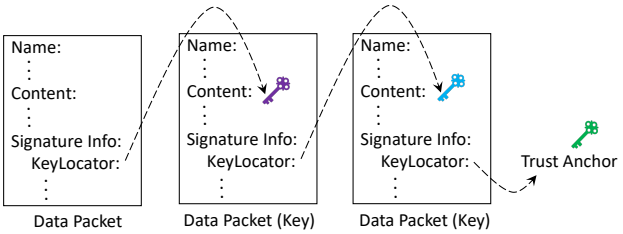


Figure 7: A key chain including the target data, intermediate keys, and the trust anchor (trusted public key).

5. Trust Management in NDN

Trust management plays an essential role in NDN by helping the data consumers overcome uncertainties and risks associated with the globally addressable and routable contents transmitted by applications and services communicating on the network. Trust relationships are established by NDN entities using the built-in data-centric security mechanism through the data producer's signature to assert data authenticity, integrity, and correctness. At the time of data creation, the producer attaches a digital signature to bind the data packet with its name for data integrity and provenance [58]. Authentication and authorization of NDN data packets are facilitated by incorporating a trust model, also known as *security context*, comprised of *trust anchors*, *trust chain*, and a set of rules.

Trust anchor is the pre-trusted public key, and the data producer stores the reference to this key in a specific field of the data packet called KeyLocator [59]. Consumers can use this field to retrieve the pre-trusted public key for data packet verification. Moreover, this field can refer to one of three things: (1) the public key, (2) a certificate containing the public key, and (3) an NDN name referencing the content that contains the public key [60], which implies that the consumer may need to retrieve multiple keys before reaching the trust anchor, as shown in Figure 7. A set of rules is required to correctly derive the trust anchor through the *trust chain*, which represents the list of keys between the target data and trust anchor. Producers and consumers of the same NDN application are required to share the same trust model. NDN allows each organization to choose its specific trust model based on its application's needs. The fundamental goal of the trust model is to establish a relationship between the data names and the legitimate keys used to sign data packets.

5.1. Trust Model

A digital signature is essential in assuring the consumer that a legitimate producer created the received content. The generalized steps for signing and verifying contents can be summarized as follows:

- Key Generation: This process generates a pair of the private key (signature key) and a public key (verification key). The generation procedure usually uses a random number generator that produces unique pairs each time.
- Signing Process: The process takes data as input, feeds it into a hash function, and outputs a digest that is a string of digits, then encrypts the digest using the private key of the producer. The result of the encryption process is the producer signature, shown in Figure 8. To attach the producer signature with data, the NDN data packet includes a *DataSignature* field which consists of two elements: *SignatureInfo* and *SignatureValue* [61]. The *SignatureValue* holds the actual bits of the signature produced through data digest encryption.

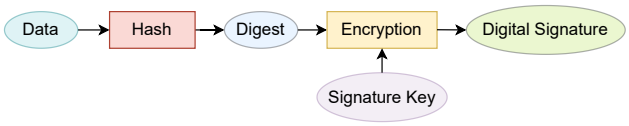


Figure 8: Generating Digital Signature

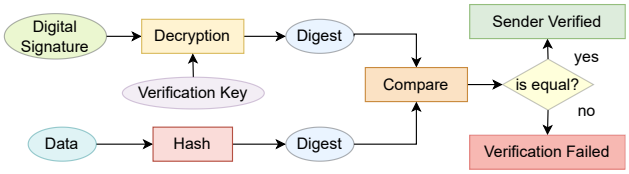


Figure 9: Verifying Digital Signature

- Verifying Signature: To validate a data producer, a consumer must verify the signature integrated with the data packet. The consumer decrypts the signature to generate a digest using the trust anchor (trusted public key), fetched through the *KeyLocator* field located at the *SignatureInfo* element of *DataSignature* field. Required information about the digital signature algorithm, such as the hash and encryption functions utilized during signature generation, is included in the *SignatureInfo* element. For a valid producer, the digest generated from the decryption should match the digest produced by hashing the data, as shown in Figure 9.

The algorithms used to produce the signature vary mainly on the techniques used to create the public and private keys and are selected by the content producer according to the need of the data or application. One of the most popular algorithms to generate the keys is the RSA (Rivest–Shamir–Adleman) algorithm, an asymmetric encryption algorithm.

5.2. Trust Management Policy

Producer signature verification helps ensure data integrity, assuming the consumer trusts the producer. However, in a highly dynamic environment such as IoV, the vehicular nodes may come in proximity for the first time and a brief period, often less than ten seconds [62]. Due to mobility, it might be challenging to trust the data generated by any vehicular node since improper or inadequate processing of the desired data or misleading messages in a safety-critical environment can have disastrous consequences. Trust management policies can help in this regard by providing guidelines to determine which producers to trust and which ones to be cautious about, ensuring secure and reliable data communication in safety-critical environments. Ramani et al. [63] propose a swift trust model for an NDN-based vehicular environment to establish trust between

vehicles that encounter each other briefly. This trust model requires nodes to establish short-term trust before requesting data for safety-critical applications, such as lane changes. Upon joining the network for the first time, a vehicle requests its neighbors to complete a simple task that it can later verify objectively or quantitatively. Based on the evaluation, which includes factors such as the promptness and precision of responses, the requesting vehicle assigns trust scores to its neighbors and establishes trust to proceed with safety application communication.

6. Challenges in NDN Adoption

This section presents a broad overview of the challenges encountered by applications that adopt NDN. The challenges are categorized based on the beneficial features of the NDN architecture for these applications, as summarized in Table 3.

6.1. Challenges with Caching Contents

While NDN allows routers to cache named contents, which improves content retrieval, it also opens up the network to several adversarial attacks. These attacks can include injecting corrupted contents into the router's CS, compromising consumer privacy, and restricting consumer access to data. Furthermore, caching contents may pose challenges for devices with limited storage resources.

6.1.1. Content Poisoning Attack

To ensure the authenticity and correctness of each content, NDN requires producers to sign content and consumers to verify the signature. However, verifying the signature of each cached content in routers can be impractical due to two main issues. First, signature verification imposes additional overhead, which requires significant processing power, particularly for routers with multiple Gigabit-speed interfaces [64]. Second, routers need to establish trust in the network to obtain the correct key for content verification. Although contents include a reference to the required key, routers must learn all the trust models used in the network to determine the authenticity of the key. This flexibility to choose any trust model allows corrupted or fake contents to be cached [65, 64].

Caching contents in routers opens the network to adversarial attacks, such as content poisoning attacks, which can inject corrupted contents into the router's cache, compromising consumers' privacy and preventing them from obtaining authentic content. The attacker can induce the router's cache with fake or corrupted contents, which are then forwarded to the consumers, achieved by controlling compromised routers or hosts distributed across the network [64]. Compromised routers can simply reply with poisoned data whenever an interest in cached content is received. This way, the requester and other intervening routers will cache the poisoned content. In the case of compromised hosts, attackers can predict interests in particular content and reply to simultaneous interests with poisoned content from compromised nodes in the network, which will be cached by routers involved in forwarding the data and returned to legitimate consumers who issue interests in the content.

6.1.2. Time Analysis Attack

A privacy breach can be caused by time analysis attacks, which attempt to determine if a user has requested specific content or not [66]. Attackers can analyze the difference between cached and uncached content to perform such attacks. The attacker who shares the victim's first-hop router (R_1) requests content (C) and logs the time (T_1) it takes to retrieve it. Then, the attacker requests the same content again and logs the time (T_2), assuming that R_1 has cached the content, and $T_2 < T_1$. Finally, the attacker monitors the victim's network traffic to obtain the fetch time and size of each content. By comparing the content's retrieve time with T_2, the attacker can infer whether the victim requested content C.

6.1.3. Bogus Announcement Attack

After any change in route or content, routers must communicate to exchange updated information to achieve new global consistency. The routing convergence period, which is the time between the occurrence of a change and when all routers agree on the new state, should not be too long as the routers need to get back to delivering the contents. However, in the bogus announcement attack, an attacker sends many announcement updates for a particular content cached in the routers' CS at a high frequency that exceeds the convergence time. Due to these quick updates, routers will not be able to address legitimate user requests leading to failure in retrieving complete or correct content [67].

6.1.4. Resource Constraints

Storing every content for extended periods requires significant storage capacity in each router, which can be costly. Given cost constraints, routers, and other nodes like vehicular sensor nodes have limited cache space. Therefore, the development of caching replacement policies becomes necessary to optimize storage resources, which in turn enhances caching while minimizing storage expenses. These policies also guarantee that necessary content is cached at the appropriate time, which minimizes data delivery latency.

6.2. Challenges for Naming Contents

Content naming is used in networking applications to enable mobility, eliminating the need to monitor IP addresses whenever a node changes its network. However, naming data poses risks to consumer privacy, as content names in interest and data packets can be monitored, potentially revealing consumer privacy. Additionally, issues may arise when checking for names every time data and interest packets are forwarded, including the name lookup during data and interest packet forwarding. In NDN, packet forwarding decisions use content names instead of IP addresses. When an interest for specific content arrives at a router, exact string matching is done in CS and PIT. If both fail, the data was not requested recently through that router. LPM is then done in the FIB to find where to forward the interest packet. Similarly, when a data packet arrives, the content name inside the data packet is matched with the PIT entries to find the interface to which the packet should be forwarded. However, content name lookup consumes time and memory.

Table 3: Summary of NDN Limitations Categorized by NDN Features

Features	Challenges	Description	Drawbacks
In-network Caching	Content Poisoning	Corrupted contents get cached as the routers in the network are not required to verify cached content due to verification overhead	Corrupted contents are forwarded to the consumers to prevent them from obtaining authentic content
	Time Analysis Attack	Attacker analyzes the content fetch time difference between cached and uncached content	Compromise users' privacy by determining whether the victim has requested a specific content or not
	Bogus Announcement Attack	At a high frequency, an attacker sends updates for a particular content cached in the targeted routers' CS	Targeted routers will not be able to address legitimate user requests due to high-frequency update requests
	Resource Constraints	Caching every content in the CS that passes through	Caching every content for a long time is expensive and challenging for resource constraint nodes
Content Naming	Name Lookup	Content can have names of variable length as there is no imposed upper bound on length	Challenging to perform fast name lookup and manage NDN name tables
	Watchlist & Sniffing Attack	Attacker monitors the victim's network traffic using a predefined list (watchlist) or randomly-guessed list (sniffing) of content names. Delete the interest packet and/or record the requesters's information if a match is found from the list	Failure in content retrieval and compromise victim's privacy
	Unencrypted Interest Packets	Interest packets are sent without encryption in cleartext	Hinders requester's privacy by enabling easy sniffing, spoofing, and capturing requested information
Routing	Dynamic Flooding	Adversary tries to block network regions by traversing neighboring networks and sending several requests to routers	The blockage leads to longer response delay by creating denial-of-service attack
	Timing Attack	Attacker overloads the network with many requests through one or more routes	Creates to delay in response leading to deletion of interests from the PIT
	Interception	The attacker acts as the trusted publisher of the requested contents to receive the interests	Requester's privacy gets violated due to the man-in-the-middle attack
	Mobility	The connectivity of the nodes in a mobile network are short-lived and dynamic	The inability to maintain connectivity in mobile networks makes it challenging for data retrieval using the exact reverse path used by interest packets, causing failure in data retrieval
Producer Signature	Verification Overhead	Routers do not verify the producer signature of the cached contents due to overhead	Leads to the risk of content poisoning attack
	Certificate Retrieval	The nodes involved in the trust chain need to be active while fetching the verification keys, which is challenging in a disruptive ad-hoc environment	The disruption in network connectivity makes the signature verification procedure challenging
	Data Packet Size	To verify producer signature, producer's public certificate is included in data packet	Challenging for resource constraint networks
	Producer's Privacy	Each data packet includes producer information to allow consumers verify producer signature	Producer anonymity is not possible to maintain
	Corrupted Trust Anchor	Trust anchor may get attacked and corrupted	Difficulty in verifying received data packet
Others	TCP/IP Integration	Differences between NDN and TCP/IP core elements	Leads to NDN's inability to cohabit alongside TCP, which disrupt seamless data transmission
	Reliability	No built-in reliability mechanism	Individual applications need to implement data recovery techniques

There are several issues related to content name lookup during forwarding interest or data packets. Firstly, NDN names have no externally imposed upper bound, and arbitrary names incur a lookup time proportional to their length, making it challenging to perform searches at wire speed. Secondly, name tables in NDN can be multiple orders of magnitude larger than today's IP routing tables, and managing them is difficult without data compression mechanisms. Thirdly, the NDN name table has a high update rate, making fast insertion and deletion necessary to deliver contents without delay. Lastly, content names in NDN are hierarchical strings composed of a sequence of components, making name lookup more complex than IP lookup.

NDN introduces some potential security concerns related to content naming and interest packets. Applications in NDN can use meaningful content names that are semantically correlated to the actual contents they represent. While this feature improves data retrieval efficiency, it may also lead to privacy risks and various types of attacks, including eavesdropping, watchlist, and sniffing attacks. Eavesdropping attacks involve adversaries monitoring a specific consumer's local activity and observing interest packets with relevant content names, potentially revealing sensitive information about the consumer's content preferences and intentions, compromising privacy and confidentiality. Watchlist attacks occur when attackers monitor a predefined list of content names, allowing them to interfere with interest packets or record requester information. Sniffing attacks involve attackers trying to guess content names to monitor network activities. Additionally, interest packets in NDN are not encrypted, making them vulnerable to sniffing, spoofing, and capturing details of the requested content, raising further privacy concerns.

6.3. Routing Challenges

NDN's routing mechanism aims to retrieve contents from the nearest source and distribute name prefixes while protecting the network against various attacks through multipath forwarding and interest aggregation mechanisms. However, there are still vulnerabilities in the network that can lead to denial of service attacks [68] and violations of requester privacy. The following challenges highlight some of these attacks:

- Dynamic Flooding: In this attack, an adversary moves in a circular path, traversing neighboring networks and flooding routers with multiple requests. The goal is not only to block a specific network but to isolate an entire network region by overwhelming mobile access routers. This blockage results in longer response delays across the network or a successful denial of service attack.
- Timing Attack: If a request exceeds its associated deadline in the PIT, it becomes invalid and gets deleted from the PIT. In a timing attack, the attacker floods the network with numerous requests through one or more routes. This process introduces longer delays in the network, causing requests to miss their deadlines and be deleted from the PIT. Additionally, users who are forced to request again contribute to the network flooding, leading to a denial of service.
- Interception: The goal of an interception attack is to gain access to the information of a requesting node in the network. In this type of attack, the attacker, posing as a valid publisher, maintains a list of authentic routes to reach contents in the network. When routers request contents, the attacker provides them with invalid routes. By acting as the trusted publisher of the required content, the attacker receives the requests and forwards them to the legitimate publisher to obtain a copy. The attacker then returns the content to the router, which eventually forwards it to the user. This man-in-the-middle attack violates users' privacy by intercepting their communication with publishers in the network.
- Mobility: Tracking the route of interest packets is crucial since data packets follow the same path in reverse for delivery in NDN. However, in mobile networks like ad-hoc networks, the dynamic network topologies pose a challenge. The mobility of routers and consumers in these networks leads to constant changes in router locations, causing intermittent connectivity. This lack of connectivity in mobile networks results in delays and failures in data retrieval, as maintaining a consistent reverse path for interest packets becomes difficult.

6.4. Producer Signature

NDN ensures data integrity, provenance, and authenticity by requiring content producers to cryptographically sign their data before transmission. However, securing each content introduces overhead due to signature verification mechanisms. The signature feature of NDN lacks flexibility in maintaining producer identities confidential, leading to the following challenges:

- Verification Overhead: Verifying every content in NDN increases network delay, impacting data delivery. It is impractical for routers to verify each content before caching and forwarding due to the computational overhead of cryptographic verification. Therefore, NDN does not mandate checking each packet at every router, but this opens up the network to the risk of content poisoning attacks.
- Certificate Retrieval: To verify the producer's signature in content, consumers may need to retrieve multiple keys through the trust chain, reaching the trust anchor. However, in disruptive ad-hoc environments with intermittent connectivity issues [69], maintaining active communication channels for fetching keys at all times is challenging. The intermittent connectivity poses obstacles to implementing signature verification procedures effectively.
- Size of Data Packets: Signature verification for each content in NDN requires including the producer's public key and certificate in each data packet. An alternative is to provide pointers for retrieving the certificate using the trust chain. However, this process presents challenges for networks with low data rates, such as mobile tactical networks used in battlefield communication.
- Producer's Privacy: In NDN, the content of each data packet contains information about the producer, improving data authenticity and allowing consumers to verify received content. However, this approach does not guarantee producer anonymity when required.
- Corrupted Trust Anchor: The security of each content relies on the trust anchor, a pre-trusted key. As the focus has shifted from securing the data channel to securing the data itself, the trust anchor becomes a potential target for attacks and corruption. A compromised trust anchor complicates both key and content verification processes.

6.5. Integration with TCP/IP

NDN has several advantages over the TCP/IP protocol. In NDN, there is no need for name-to-IP translation or vice versa since NDN directly names the content. Additionally, NDN includes built-in privacy considerations, which are lacking in the IP network design. Unique features of NDN, such as in-network caching and interest aggregation, enhance the reliability of NDN networks. However, TCP/IP has become the predominant technology for connecting network devices, making it necessary for NDN to coexist with TCP/IP backbone networks for seamless transmission across the network. Unfortunately, there are significant differences between NDN and TCP/IP that make this coexistence challenging, particularly in routing and forwarding mechanisms.

In IP networks, routers exchange data and generate forwarding tables to determine the best route at the routing plane. The forwarding plane then delivers data based on the information in the forwarding table, resulting in a stateless forwarding plane. In contrast, NDN has a stateful forwarding plane, allowing routers to forward packets along the exact path taken by interest packets. NDN routers maintain the state of pending interests to guide data packets through the correct routes.

Incorporating unique features of NDN, such as in-network caching, into the existing TCP/IP infrastructure presents challenges. It is also difficult to handle multiple interests from large IP networks with a single data packet, which can lead to protocol conversion overheads and impact data transmission.

6.6. Reliability

Reliability is critical to network communication, ensuring data loss prevention, seamless connectivity, and the ability to recover from losses caused by various network conditions. In the context of NDN, where data retrieval is based on content names rather than IP addresses, achieving reliability presents unique challenges, especially in highly dynamic networks like the IoV. In NDN, the responsibility for making forwarding decisions and ensuring reliable data delivery lies with the strategy layer. When a data packet is requested but not received within the retransmission timeout timeframe, the requester will retransmit the interest packet. However, in IoV environments, network participants, such as vehicles, often join and leave the network abruptly due to mobility and changing network conditions. This can result in frequently disrupted and disconnected links between requesters and data senders.

One fundamental challenge in NDN is the absence of a built-in mechanism to establish a reliable end-to-end connection in situations where intermittent nodes and data senders may intermittently join and leave the network. While NDN does offer a naive solution called multi-path transmission to enhance the probability of successful packet delivery, it is important to note that this approach can introduce significant overhead and network congestion due to redundant data propagation. For example, in IoV scenarios, vehicles may enter and exit network coverage areas frequently, leading to unpredictable connectivity. Data loss in such situations can have significant consequences, ranging from delayed traffic updates to compromised safety-critical communications. Thus, achieving reliability in NDN, particularly in dynamic and intermittently connected environments like IoV, remains a complex and ongoing challenge.

7. Emerging NDN Strategies

In this section, we discuss some of the proposed solutions that address the challenges faced by applications adopting NDN. The approaches are summarized in Table 4.

7.1. Methodologies for Efficient and Secure Caching

Different approaches have been proposed to address caching-related challenges in NDN. Yu et al. [70] proposed a caching policy to improve cache hit rate in NDN by selectively caching popular contents on routers close to the requester. Instead of caching the requested content in every router the data packet passes through, the first-hop router tracks the frequency of content requests from consumers and attaches a count to the transmitted interest packet. Based on this count, the producer determines whether and where to cache the content. This approach reduces latency in NDN by caching frequently accessed content in close proximity to the user, while also reducing storage requirements in the CS.

Hou et al. [71] propose an innovative cache placement algorithm called the graph neural network-gain maximization (GNN-GM) to enhance user experience by increasing the cache hit rate of videos. Unlike Yu et al. [70], the authors argue that past popularity does not strongly predict the future popularity of videos. To address this, the authors leverage inductive matrix completion [95] technique, which is based on Graph Neural Network, to predict user ratings for unwatched videos. The cache replacement policy is implemented based on the rating of the videos, with higher-rated videos replacing lower-rated videos.

Im and Kim [72] introduced the use of self-certifying names to reduce verification overhead in routers and mitigate cache poisoning. A self-certifying name is created by adding a hash value to the content name as a suffix during data packet generation. If the consumer knows the hash value computed using the content, its name, and signature beforehand, it can include this value in the interest packet. Routers can then verify the authenticity of the content by comparing the hash values in the interest and data packets before making forwarding decisions.

Hu et al. [73] proposed a lightweight mechanism to enhance security in NDN caching by preventing fake content from entering the network. This approach employs name-key-based forwarding, where a consumer specifies the desired producer in the interest packet using the producer's public key. This helps reduce the chances of receiving poisoned content. If poisoned content still reaches the consumer, the consumer can reissue another interest packet, taking advantage of NDN's multi-path forwarding and invoking on-demand signature verification at intermediate routers. The reissuance of interest packets also aids in removing fake content from nodes and finding alternative forwarding options for legitimate content.

Summary and Open Research Problems: Efficient NDN caching results from various strategies addressing caching challenges. For example, selective caching places popular content closer to users, effectively reducing latency and minimizing storage needs. Additionally, deep learning algorithms, like GNN-GM, predict user ratings for videos, leading to improved cache replacement policies and enhanced user experience. Moreover, using self-certifying names and lightweight security measures ensures content authenticity, mitigates cache poisoning, and strengthens network security while facilitating improved content delivery. As a result, NDN caching optimizes resource allocation, reduces network overhead, and significantly enhances content delivery efficiency.

However, challenges remain that need to be addressed, such as caching popular content close to the requester ensure faster content delivery, but it may introduce security challenges. Moreover, obtaining producer hash values in highly mobile networks can be challenging in cache poisoning mitigation techniques, such as using self-certifying names and name-key-based forwarding. Additionally, the preference for specific producers, as mentioned in Hu et al.'s work [73], may impact the feasibility of in-network caching and potentially lead to network congestion. Future research should focus on developing efficient caching schemes that consider node mobility and ensure network security.

7.2. Naming Schemes

Several solutions have been proposed to address the name lookup challenge in NDN. Zhang et al. [74] introduced the ternary content-addressable memory (TCAM) for fast name

Table 4: Summary of Proposed NDN Strategies.

Strategies	Authors	Objectives	Methods	Limitations
Efficient and Secure Caching	Yu et al. [70]	To reduce redundant data and improve cache hit rate	Content popularity and router level based caching	Security threats are not considered
	Hou et al. [71]	To improve cache hit rate for videos	Predict user ratings for unwatched videos	Strategy may only work for videos
	Im and Kim [72]	To reduce verification overhead and cache poisoning	Self-certifying name	Getting producer hash beforehand in highly mobile network is difficult
	Hu et al. [73]	To reduce poisoned content	Multi-path and name-key-based forwarding	Adversary may flood network with reissued interests invoking router signature verification.
Naming Schemes	Zhang et al. [74]	To enable fast name lookup	Ternary content addressable memory (TCAM) to store data	Limited memory capacity of TCAM
	Wang et al. [75]		Parallel name lookup using name prefix tree	Nodes having duplicate entries result in-memory wastage
	Wang et al. [76]	To solve variable name length issue	Fixed length name codes, state transition array	The trie implementation used has memory efficiency issues.
	Wang et al. [77]	To enable name lookup at wire speed	Trie-based multiple aligned transition array data structure to construct the name table	Resource constraint network with limited GPU capacity has not been considered
	DiBenedetto et al. [78]	To address consumer privacy issue	Multiple layers of encryption for interest and data packets	No privacy guarantee in presence of a global eavesdropper
	Kaur et al. [79]	To reduce the PIT search time issue	Additional Stable Bloom Filter along with PIT	Introduces false positive rate
Routing Approaches	Wang et al. [26]	To recover from the best route failure	Extension of OSPF allowing consumer choose best path from multiple paths	OSPFN does not support dynamic multipath routing
	Tortelli et al. [80]	To reduce network overhead	Replacing FIB with state bloom filters	False positive rate not considered
	Berto et al. [81]	To provide faster lookup and efficient memory	Replacing FIB with spatial bloom filters (SBF)	Performance comparison with other SBF approaches is not mentioned
	Mick et al. [82]	To address scalability and security issues	Lightweight authentication framework for hierarchical routing	Authentication overhead due to node mobility has not been considered
	Yang el al. [83]	To address blackhole attack	Reputation-based routing scheme	Does not consider blackhole attack caused in energy-constrained IoT environment
Lightweight Signing Mechanisms	Li et al. [84]	To overcome verification overhead issues and control data access	Merkle hash tree algorithm to generate tokens	Security threats, such as token corruption is not considered
	Huang et al. [85]	To reduce verification overhead and protect producer anonymity	Edge computing devices, batch verification	Huge verification overhead despite adopting batch verification
	Ghali et al. [60]	To enable efficient certificate retrieval	Inserting public key (PK) in KeyLocator and PK hash in PPKD field	PPKD field has been removed from the current NDNv0.3
	Lou et al. [86]	To reduce trust anchor corruption	Blockchain-based key management scheme	Additional overhead due to blockchain operations in resource constraint devices
	Chatterjee et al. [87]		Merkle Patricia trie data structure	Space complexity has not been analyzed for the data structure
Integrating NDN with TCP/IP	Albalawi et al. [88]	To ensure NDN runs over IP	Overlay approach to encapsulate NDN packets into IP datagrams	IP applications can not run over NDN
	Shannigrahi et al. [89]	To enable IP application run over NDN	Underlay approach to encapsulate IP packets into NDN interests	NDN can not run over IP
	Wu et al. [90]	To support both IP and NDN traffic in an integrated network	Hybrid approach of Ethernet and NDN-enabled dual-stack switches to forward both IP and NDN traffic	IP and NDN integration in wide area network is not considered
Data Reliability	Lin et al. [91]	To overcome disconnecting link issues	Relative velocity of neighbors	Approach is limited to the nodes with active GPS
	Burhan et al. [92]		Relative velocity of neighbors and calculating forwarder timer	
	Rezaeifar et al. [93]	To construct a reliable and secure data delivery path	Reliability metric to rank router interface	Node mobility is not considered
	Lai et al. [94]	To enhance the QoS for live audio and videos in UAV	Interest packet retransmission and forwarding control strategy	Experiment only includes audio and video aspects of UAV swarm

lookup. In this approach, data names from incoming interest packets are directly loaded into the TCAM, and successive lookups match these names with their corresponding data contents. While TCAM provides fast name lookups, its drawback lies in its limited memory capacity, which can increase the name lookup cost.

Wang et al. [75] proposed the parallel name lookup (PNL) based on the Name Prefix Tree (NPT). PNL speeds up NDN lookup by utilizing memory resources in parallel. When an interest arrives at an NPT node, the name prefix lookup traverses from the root to the leaf nodes, and this process is repeated for each interest packet. When multiple interest prefixes arrive at the NPT simultaneously, all memories are visited concurrently, resulting in high-speed lookup. However, PNL may lead to memory wastage due to duplicate entries.

Wang et al. [76] proposed a name component encoding strategy to address the fast name lookup challenge and used a code allocation method to produce unique 32-bit long codes for each

content name to solve the variable name length issue. The authors suggested using state transition arrays to speed up the longest prefix match. Similarly, Wang et al. [77] present a GPU-based lookup engine that implements large-scale name lookup at wire speed using a trie-based multiple aligned transition array data structure to construct the name table exploiting the massive parallel processing power of GPUs. To address the fast name lookup challenge, Wang et al. [76] suggested a name component encoding strategy and employed a code allocation method to assign unique 32-bit long codes to each content name, resolving the issue of variable name lengths. The authors also proposed the use of state transition arrays to speed up the longest prefix match. Similarly, Wang et al. [77] presented a GPU-based lookup engine that achieves large-scale name lookup at wire speed. They utilized a trie-based multiple-aligned transition array data structure to construct the name table, taking advantage of the massive parallel processing power of GPUs.

Kaur et al. [79] proposed a Stable Bloom Filter (SBF) based PIT (S-PIT) to reduce the PIT search time by efficiently identifying queried content using an additional SBF data structure. The PIT search time tends to become higher as its size increases with the addition of new content names. However, SBF is a Bloom filter of fixed size m, representing each element using d bits. Therefore, in S-PIT, incoming interest packets are first checked against SBF using a hashing-based search before performing a direct search in the PIT, which could contain millions of entries.

DiBenedetto et al. [78] proposed an anonymous named data networking application (ANDaNA) that works as an anonymization tool for NDN to address consumer privacy and prevent adversaries from linking consumers with the content they are retrieving. ANDaNA employs multiple layers of encryption for interest and data packets. These packets are then routed through a sequence of anonymizing routers, where each router successively removes a layer of encryption and forwards the decrypted messages to the next hop. This meticulous process guarantees anonymity and unlinkability of the contents with consumers, ensuring their privacy remains intact throughout the data retrieval.

Summary and Open Research Problems: The efficiency in NDN name lookup results from several methods proposed to address the name lookup challenges and optimize content retrieval in NDN networks. Introducing strategies such as TCAM for fast name lookup, PNL utilizing memory resources in parallel, name component encoding with code allocation, and SBF-based PIT has significantly improved name lookup speed, while ANDaNA ensures privacy and secure data retrieval. These approaches enhance NDN's efficiency by streamlining name-based content access and protecting user privacy.

However, the solutions mentioned above have certain limitations that require further consideration. For example, the TCAM approach mentioned in Zhang et al.'s work [74] has limited memory capacity, which can impact scalability. PNL may result in memory wastage due to duplicate entries. Furthermore, the name component encoding strategy assigns fixed-length names to each content, which limits flexibility. Additionally, the GPU-based lookup engine proposed by Wang et al. is unsuitable for resource-constrained devices lacking GPU power. Therefore, future research should address these limitations and optimize the proposed solutions to enhance the efficiency and scalability of the naming mechanism in NDN.

7.3. Routing Approaches

NDN routing mechanisms primarily focus on selecting the best path, ensuring the availability of multiple routes to the destination, and implementing efficient forwarding mechanisms to provide stability and scalability [22]. In line with these objectives, various approaches have been proposed that offer the shortest path based on name prefixes.

Open shortest path first for named data (OSPFN) [26] is an extension of the widely used IP-based routing protocol called open shortest path first (OSPF) [96], which only provides a single path. However, NDN supports multipath forwarding, and OSPFN allows consumers to specify an alternative path when the best path fails to retrieve the data. OSPFN uses opaque link-state advertisements to broadcast name prefixes and constructs the best route. This approach is also backward compatible with nodes using IP.

Some approaches consider replacing the FIB with improved lookup tables using bloom filters. Tortelli et al. [80] introduced an intra-domain content-driven routing algorithm that reduces network overhead using state bloom filters (SBF). This approach utilizes SBF to handle link failure and supports link recovery, making it a preferable alternative to flood-based routing. Similarly, Berto et al. [81] suggested replacing FIB with a spatial bloom filter that employs hashing principles to ensure fast lookup and efficient memory consumption. Mick et al. [82] proposed a lightweight authentication framework that supports hierarchical routing in NDN to address scalability and security concerns. The framework ensures secure onboarding of every node in the network using pre-shared keys and an authentication manager.

Yang el al. [83] proposed a reputation-based scheme called SmartDetour, which consists of two key components: the reputation-based probabilistic forwarding strategy (RPFS) and the proactive reputation updating algorithm (PRU). This scheme is designed to tackle blackhole attacks, where malicious routers drop interest packets during forwarding. When an interest drop occurs along a path, it affects the reputation of all routers on that path. However, SmartDetour's PRU efficiently isolates the attacker router from the route while updating the reputation of other routers, ensuring that the reputation values accurately reflect their trustworthiness. With the RPFS component, each node can intelligently select the next hop based on the reputation parameter, enhancing the overall security and reliability of the NDN network.

Summary and Open Research Problems: The achieved efficiency of NDN routing strategies is attributed to selecting the best path and ensuring multiple routes' availability to the destination, such as OSPFN, which allows consumers to specify alternative paths when the best one fails. Strategies with bloom filters improve lookup tables and handle link failure efficiently, reducing network overhead and supporting link recovery. Lightweight authentication frameworks ensure secure on-

boarding on each node in the network, while reputation-based schemes, such as SmartDetour, enhance security and reliability by intelligently selecting the next hops based on reputation parameters. By adopting these routing approaches, NDN optimizes data delivery, enhances routing stability, and provides a scalable and efficient routing infrastructure for modern communication networks.

However, the stated routing methods have research gaps that need further exploration. For instance, in OSPFN, there is a need to investigate dynamic mechanisms for handling route failures without involving the requester, thereby preventing malicious nodes. An adaptive algorithm could be designed to dynamically reroute traffic based on network conditions, ensuring efficient and secure data delivery. Furthermore, the approach proposed by Tortelli et al. [80] focuses solely on intra-domain content-driven routing, prompting the exploration of extending SBF for inter-domain routing. Additionally, the methodologies proposed by Berto et al. [81] and Mick et al. [82] should be further evaluated in a large-scale and highly mobile NDN environment to assess their effectiveness.

7.4. Lightweight Signing Mechanisms

One of the challenges in incorporating signatures in NDN is the overhead associated with verification. To address this challenge, solutions using lightweight and efficient verification methods have been proposed. Li et al. [84] proposed an integrity verification architecture that allows content providers to control data access using a lightweight hash algorithm called the Merkle hash tree algorithm [97] to generate tokens for signature generation. To verify the signature, the consumer sends an interest to request the token, and the content provider reviews the public key to determine if the consumer is authorized to receive a valid token.

Huang et al. [85] introduced an architecture that shifts the signature generation process to edge computing devices. The authors propose batch verification to verify multiple data packets and reduce the verification overhead. To protect the anonymity of producers, the authors implement a group signature method in which any group member can sign content on behalf of the group. Ghali et al. [60] proposed the Interest-key binding rule, in which the producer adds the public key to the KeyLocator field of the data packet, eliminating the need to fetch keys from the trust chain. The consumer obtains the producer's public key before sending an interest packet and includes the digest of the public key in the optional ProducerPublicKeyDigest (PPKD) field of the interest packet. When an interest is received, the intermediate routers fetch the public key from the KeyLocator, compute the digest, and compare it with the PPKD value. The routers accept the content if the digests match and discard it if there is no match.

Lou et al. [86] proposed a blockchain-based key management scheme for NDN to address trust anchor corruption. The proposed approach stores and distributes the hash values of pre-trusted public keys across blockchain nodes, making it difficult to tamper with. Moreover, the authors changed the data packet format by excluding the KeyLocator field and including a block height and transaction ID. Using the block height and transaction ID, the consumer can retrieve the stored hashed value from the blockchain and match it with the PPKD field. However, blockchain incurs additional overhead for verifying stored data on resource-constrained devices, and this work relies on the assumption that blockchain can be deployed in NDN. In contrast, Chatterjee et al. [87] proposed a cryptographically secured data structure, Merkle Patricia Trie [98], to keep logs of public keys. The authors suggest monitoring the logs to detect any suspicious changes in transmitted data.

Summary and Open Research Problems: Various proposed approaches aim to achieve efficient and lightweight signing mechanisms in NDN, minimizing signature verification overhead and ensuring secure and reliable content distribution. For instance, lightweight and efficient verification methods, such as the Merkle hash tree algorithm, enable content providers to control data access using tokens for signature generation. Additionally, shifting the signature generation process to edge computing devices, as proposed by Huang et al. [85], reduces verification overhead through batch verification and group signature methods. Moreover, the Interest-key binding rule eliminates the need to fetch keys from the trust chain by including public key digests in the interest packet. Furthermore, blockchain-based key management schemes ensure trust anchor integrity by storing hash values of pre-trusted public keys on blockchain nodes. These advancements in data packet signing enhance the security and reliability of NDN communication while minimizing computational burden and verification complexity.

However, there are still some research gaps that need to be addressed. For example, the methods proposed by Li et al. [84] and Ghali et al. [60] assume that the content producer is trusted, but malicious producers may violate this assumption, leading to potential security threats. Additionally, the batch verification approach suggested by Huang et al. [85] did not consider the offloading time to edge servers in real-time systems. Furthermore, Lou et al. [86] and Chatterjee et al. [87] did not evaluate the overhead of verifying and monitoring stored data on resource-constrained devices. Addressing these challenges will provide solutions that improve the security and reliability of NDN.

7.5. Integrating NDN with TCP/IP

The integration of NDN and TCP/IP allows for the coexistence of these two network architectures and enables efficient communication between nodes using these architectures. Conti et al. [99] propose three configurations for integrating NDN and TCP/IP: overlay, underlay, and hybrid techniques. The overlay approach uses a tunnel over the IP protocol to ensure NDN runs over IP, while the underlay solution enables NDN to run under IP by using protocols, proxies, and conversion gateways to efficiently send and receive requests between NDN and IP nodes. The hybrid approach adopts a dual-stack node that handles the semantics analysis of both IP and NDN packets [100].

Albalawi and Garcia-Luna-Aceves [88] propose an overlay approach called the Named-Data Transport Protocol (NDTP) that encapsulates packets into UDP datagrams, which are further encapsulated into IP datagrams. This approach allows an

NDN network to run over an existing TCP/IP network. The implementation allows the producer and consumer to share a description of shared content using a manifest file that contains a record describing the location and structure of shared content. Additionally, the manifest is made available to the underlying IP network with details to ensure efficient retrieval of these contents.

Shannigrahi et al. [89] propose IPoC, an underlay configuration that enables IP applications to run over NDN. IP packets are encapsulated in NDN Interests by an IPoC client and sent to the backbone NDN network. Messages coming into the IP network are received at the edge by an IPoC gateway, which unpacks the IP packets before sending them to the appropriate network. This approach ensures that network infrastructures can continue to run IP over an underlying NDN architecture until the complete transition from IP to NDN occurs.

Wu et al. [90] utilize Ethernet and NDN-enabled dual-stack switches that support both NDN and IP traffic to facilitate IP and NDN applications. In the hybrid approach used by the authors, each switch in the network builds its MAC-address-based switching table to forward IP packets. Furthermore, Ethernet switches forward NDN packets based on the associated MAC addresses of the packets, while dual-stack switches can do so based on the packets' content names. Using dual-stack switches alongside Ethernet switches ensures that both IP and NDN architectures can interwork to create an integrated network that facilitates the forwarding of IP and NDN packets.

Summary and Open Research Problems: NDN and TCP/IP network integration has been facilitated through various techniques, including overlay, underlay, and hybrid approaches. These strategies enable efficient coexistence and communication between the two architectures. For instance, the overlay approach uses tunnels over the IP protocol, while the underlay solution employs protocols and proxies to exchange requests between NDN and IP nodes efficiently. Additionally, the hybrid approach utilizes dual-stack nodes to handle the semantics of both IP and NDN packets. By adopting these integration methods, the networks can efficiently run IP over NDN or NDN over IP, ensuring seamless data exchange and improved communication efficiency.

However, there are research gaps in each approach that need to be addressed, such as further testing, scalability, and optimizations to accelerate the deployment and practical usage of these techniques. For example, the overlay approach proposed by Albalawi and Garcia-Luna-Aceves does not enable IP applications to run over NDN, while the method proposed by Shannigrahi et al. does not authorize NDN applications to run over IP. Furthermore, although Wu et al. attempt to address these challenges, the integration was not evaluated in wide area networks. Overall, the integration of NDN and TCP/IP is an ongoing area of research with promising potential for scalable and efficient network architectures.

7.6. Approach for Data Reliability

Different approaches have been explored to address the challenges of reliable data transfer in NDN [91, 92, 93]. Lin et al. [91] proposed a reliable forwarding strategy to address the non-delivery of data caused by disconnecting return paths or node mobility. In their scheme, an NDN node considers the relative velocity of its neighbors when forwarding an interest. This method selects the next node in such a way that the data packet can return to the consumer over the same reverse path. Similarly, Burhan et al. [92] addressed the issue of disconnecting links during data packet delivery. In addition to considering vehicular mobility, the authors suggested a forwarder timer that each node calculates after sending Interests. If the data packet does not reach the node before the timer expires, the node will re-send the interest packet, assuming it was lost in the network.

Rezaeifar et al. [93] propose a reliable adaptive forwarding approach that takes into consideration unreliable and unauthorized routers in the network. The proposed method uses a reliability metric to rank the router interfaces based on reputation, allowing the selection of the next interface to forward the interest and establish a reliable data delivery path.

Lai et al. [94] present a proposal to enhance the quality of service (QoS) for live audio and video streaming transmission in UAV swarm networks by replacing TCP/IP with NDN. To improve transmission reliability, the authors introduce two strategies: an interest packet retransmission control strategy and an interest packet forwarding control strategy. The retransmission control strategy dynamically adjusts interest packets' retransmission timeout at the consumer node, enhancing network performance by decreasing the transmission failure ratio. The forwarding control strategy dynamically adjusts the interest packet forwarding time interval based on the number of neighboring nodes of the next hop. By slowing down or speeding up forwarding based on node density, this strategy reduces conflicts and contributes to overall network performance improvement.
Summary and Open Research Problems: Various strategies have been developed to enhance data reliability in NDN networks, addressing challenges such as packet loss, disconnecting links, and unreliable routers. For instance, reliable forwarding strategies consider factors like node mobility and relative velocity of neighbors to select optimal paths for data packet return. Adaptive forwarding approaches use reputation-based metrics to rank router interfaces, enabling the selection of reliable routes for data transmission. Furthermore, interest packet retransmission and forwarding control improve the quality of service for live audio and video streaming. By employing these data reliability enhancement techniques, NDN networks can achieve more dependable and efficient data communication, making them well-suited for real-world applications. However, further optimization is required to address research gaps, including the consideration of time-critical applications and high node mobility.

While NDN offers promising potential as a seamless communication architecture, it faces challenges related to caching, forwarding, security, and signature incorporation. Exploring existing networking architectures to identify beneficial concepts and implementing them to overcome these challenges in NDN is an active research domain that networking experts can explore.

8. Open Research Challenges and Future Directions

While NDN has demonstrated great potential in various application domains, including content distribution, IoT, IoV, and edge computing, its deployment and adoption face several challenges. Despite significant progress in addressing various challenges of NDN, developing novel approaches that facilitate the seamless adoption of NDN in current Internet applications remains an ongoing research area. Furthermore, as NDN technology advances and new use cases emerge, there is a growing need for innovative solutions to overcome the remaining barriers to its widespread adoption. Therefore, in this section, we highlight some of the open research challenges that need to be addressed and provide recommendations and suggestions for future research directions to tackle these challenges.

8.1. Scalability

As the number of connected devices and data traffic volume continue to grow, ensuring the scalability of NDN technologies becomes critical. To support a high link rate, such as 10 Gbps link capacity or data transfer rate between NDN nodes, the NDN forwarding and routing plane must efficiently scale to process tens of thousands of Interest and Data packets per second [21]. However, as more data is generated and requested, there is a growing need to cache larger amounts of data for efficient data retrieval, leading to storage challenges in resource-constrained devices. One specific challenge arises in the name lookup process, where effective solutions like trie-based algorithms may face degradation as more names are inserted into the NDN router. The lookup time can become dependent on the trie depth, impacting the overall efficiency of the routing process [101].

To address these scalability challenges, future research efforts in NDN should focus on developing robust, reliable, and scalable solutions. These solutions should include adaptive routing strategies, efficient caching mechanisms, and fast name lookup algorithms that can handle the increasingly diverse data traffic and dynamic network patterns while maintaining low latency, high throughput, and reliable performance. Additionally, exploring interdisciplinary collaborations in areas such as distributed systems, security, and machine learning will be essential to advance the state-of-the-art in NDN scalability and enable its widespread adoption across multiple application domains. By addressing these scalability issues, NDN can effectively accommodate the growing number of nodes and data traffic, ensuring its suitability for the diverse and dynamic network patterns of the future.

8.2. Real-time Communication

NDN's best-effort routing approach does not prioritize traffic based on time-criticality, which can result in data delivery delays and congestion, particularly for time-sensitive applications like IoT smart health, industrial automation, and IoV. Researchers are exploring various techniques to address this challenge, such as efficient caching and routing methods to reduce data delivery time. However, due to the lack of prioritization, time-critical data may still experience delays caused by frequent non-critical data. To tackle this challenge, we have proposed an approach using a critical deadline first scheduler inside the NDN router [102]. This technique allows packets in the router buffer to be scheduled based on their deadline. Nonetheless, further research is needed to evaluate the effectiveness of these approaches and develop new solutions that specifically address the unique challenges of NDN's data delivery in safety-critical applications.

8.3. Security

While NDN provides content-centric security through cryptographic content signatures, the system is still vulnerable to various security threats, such as cache poisoning attacks. This vulnerability arises because the routers used for in-network caching do not verify the content signature before storing it. Moreover, NDN's naming conventions for data are based on content attributes, and the interest packets are not encrypted, which can be exploited by adversaries to compromise privacy and capture information about the data, requester, and producer. Additionally, adversaries can intentionally drop packets to save their device energy, especially in resource-constrained IoT networks where NDN is used. To address this challenge, we have proposed a reputation-based forwarding approach with a reactive reputation updating mechanism [103]. However, further research is needed to detect malicious nodes that may transmit incorrect information during reputation calculation in the proposed method. Therefore, future research in NDN security should focus on developing more robust mechanisms for in-network caching, exploring encryption techniques for interest packets, improving data naming conventions, and detecting misinformation to enhance the overall security and privacy of the system.

8.4. Trust Management

Trust management plays a crucial role in facilitating secure communication between data producers and consumers in the NDN network, minimizing the risk of security breaches. To effectively support various network entities, trust management schemes need to address key challenges such as managing keys efficiently, dealing with untrusted routing paths, optimizing resource utilization, and minimizing energy consumption. In addition to these challenges, trust management schemes should incorporate efficient procedures for key rollover and revocation to establish and maintain trust relations among participating nodes. This ensures that trust is maintained even in dynamic network environments where nodes may join or leave the network. In mobile networks, trust management becomes even more critical. Efficient trust management approaches are required to establish short-term trust among moving nodes with intermittent connectivity. These solutions need to address the unique challenges posed by mobility, intermittent connectivity, and resource limitations. Such trust management schemes are essential to support safety-critical applications and enable trusted and efficient communication in NDN.

By developing efficient and robust trust management schemes, NDN can ensure secure and reliable communication,

allowing data producers and consumers to interact with confidence while mitigating security risks in the network.

8.5. Integration with Other Network Technologies

Integrating NDN with established networking technologies like TCP/IP, 5G, and 6G is crucial for facilitating the harmonious coexistence of diverse network architectures and enabling efficient communication among nodes utilizing these architectures. To achieve this integration, it is necessary to develop new routing and forwarding protocols that can effectively handle the heterogeneity of network technologies and provide support for various NDN nodes. Furthermore, there is potential for enhancing the performance and security of IoT networks by exploring the integration of NDN with emerging technologies such as blockchain and artificial intelligence. This exploration can lead to innovative solutions that leverage the unique capabilities of these technologies to further optimize the overall functionality, efficiency, and protection of IoT networks operating in conjunction with NDN.

In summary, integrating NDN with established networking technologies requires the development of specialized protocols, while also considering the potential benefits of integrating NDN with emerging technologies. These advancements hold the promise of enhancing the performance, security, and potential applications of IoT networks.

8.6. Incorporating Other Network Solutions

NDN differs from other networks due to its distinctive features, such as naming, caching, and content-based security. However, a potential research direction might be investigating the similarities between NDN and other networks, such as TCP/IP and software-defined networks, to identify beneficial concepts that can be implemented to overcome challenges in NDN. For example, similarities between NDN's hierarchical naming and IP address can help researchers leverage IP spoofing solutions to address interest spoofing. In addition, exploring the possibilities of incorporating the deadline-awareness property of time-sensitive networking might open opportunities to make NDN robust for real-time communication. By drawing insights from these established networking paradigms, researchers can gain potential solutions to enhance NDN's scalability, security, and efficiency. Therefore, exploring existing networking architectures to identify beneficial concepts and implementing them to overcome these challenges in NDN is an active research domain that networking experts can explore.

9. Conclusion

The adoption of the NDN architecture offers promising solutions to address the challenges posed by traditional TCP/IP networks and meet the communication demands of modern applications. Throughout this paper, we have presented the essential functionalities of the NDN architecture and highlighted its potential use cases. We have discussed the limitations of current network architectures and explained how NDN overcomes some of these limitations through its content-centric approach. Additionally, we have examined the drawbacks of adopting NDN technology, such as the overhead in signature verification and the potential for malicious activities by remote adversaries. To overcome these challenges, further research is required to optimize the performance of NDN and develop robust mechanisms for security and privacy protection. The integration of NDN into existing network applications holds great promise, but it demands continued exploration and innovation to fully realize its potential.

References

[1] T. Dierks, E. Rescorla, The transport layer security (tls) protocol version 1.2, RFC 5246, August (2008).

[2] E. Rescorla, N. Modadugu, Datagram transport layer security version 1.2, RFC 6347, January (2012).

[3] T. Chuachan, S. Puangpronpitag, et al., Solving mtu mismatch and broadcast overhead of ndn over link-layer networks, Ph.D. thesis, Mahasarakham University (2018).

[4] M. Buragohain, S. Nandi, Demystifying security on ndn: A survey of existing attacks and open research challenges, in: The" Essence" of Network Security: An End-to-End Panorama, Springer, 2021, pp. 241–261.

[5] L. Zhang, D. Estrin, J. Burke, V. Jacobson, J. D. Thornton, E. Uzun, B. Zhang, G. Tsudik, kc claffy, D. Krioukov, D. Massey, C. Papadopoulos, T. Abdelzaher, L. Wang, P. Crowley, E. Yeh, Named data networking (ndn) project, 2010 – 2011 progress summary, Named Data Networking.

[6] A. Robinson, Study shows how the internet's architecture got its hourglass shape, Georgia Tech `https://rh.gatech.edu/news/69297/study-shows-how-internets-architecture-got-its-hourglass-shape` (2011).

[7] V. Jacobson, J. Burke, L. Zhang, T. Abdelzaher, B. Zhang, kc claffy, P. Crowley, J. A. Halderman, C. Papadopoulos, , L. Wang, Named data networking: Executive summary, Named Data Networking `https://named-data.net/project/execsummary/` (2010).

[8] L. Zhang, A. Afanasyev, J. Burke, V. Jacobson, K. Claffy, P. Crowley, C. Papadopoulos, L. Wang, B. Zhang, Named data networking, ACM SIGCOMM Computer Communication Review 44 (3) (2014) 66–73.

[9] D. Saxena, V. Raychoudhury, N. Suri, C. Becker, J. Cao, Named data networking: a survey, Computer Science Review 19 (2016) 15–55.

[10] L. Zhang, D. Estrin, J. Burke, V. Jacobson, J. D. Thornton, E. Uzun, B. Zhang, G. Tsudik, kc claffy, D. Krioukov, D. Massey, C. Papadopoulos, P. Ohm, T. Abdelzaher, K. Shilton, L. Wang, E. Yeh, P. Crowley, Named data networking (ndn) annual report 2011-2012, Named Data Networking.

[11] Named data networking next phase (ndn-np) proposal, Named Data Networking `https://named-data.net/publications/techreports/ndn-np_proposal/` (2018).

[12] H. B. Abraham, P. Crowley, Controlling strategy retransmissions in named data networking, in: 2017 ACM/IEEE Symposium on Architectures for Networking and Communications Systems (ANCS), IEEE, 2017, pp. 70–81.

[13] V. Jacobson, J. Burke, L. Zhang, B. Zhang, K. Claffy, C. Papadopoulos, T. Abdelzaher, L. Wang, J. A. Halderman, , P. Crowley, Named data networking next phase project may 2014 – april 2015 annual report, Named Data Networking (2016).

[14] G. Xylomenos, C. N. Ververidis, V. A. Siris, N. Fotiou, C. Tsilopoulos, X. Vasilakos, K. V. Katsaros, G. C. Polyzos, A survey of information-centric networking research, IEEE communications surveys & tutorials 16 (2) (2013).

[15] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher, B. Ohlman, A survey of information-centric networking, IEEE Communications Magazine 50 (7) (2012) 26–36.

[16] X. Jiang, J. Bi, G. Nan, Z. Li, A survey on information-centric networking: rationales, designs and debates, China Communications 12 (7) (2015) 1–12.

[17] C. Fan, S. Shannigrahi, C. Papadopoulos, C. Partridge, Discovering in-network caching policies in ndn networks from a measurement perspec-

tive, in: Proceedings of the 7th ACM Conference on Information-Centric Networking, 2020, pp. 106–116.
[18] Z. Li, Y. Xu, B. Zhang, L. Yan, K. Liu, Packet forwarding in named data networking requirements and survey of solutions, IEEE Communications Surveys & Tutorials 21 (2) (2018).
[19] A. Tariq, R. A. Rehman, B.-S. Kim, Forwarding strategies in ndn-based wireless networks: A survey, IEEE Communications Surveys & Tutorials 22 (1) (2019) 68–95.
[20] M. M. S. Soniya, K. Kumar, A survey on named data networking, in: 2015 2nd International Conference on Electronics and Communication Systems (ICECS), IEEE, 2015, pp. 1515–1519.
[21] H. Yuan, T. Song, P. Crowley, Scalable ndn forwarding: Concepts, issues and principles, in: 2012 21st International Conference on computer communications and networks (ICCCN), IEEE, 2012.
[22] W. T. Ariefianto, N. R. Syambas, Routing in ndn network: A survey and future perspectives, in: 2017 11th International Conference on Telecommunication Systems Services and Applications (TSSA), IEEE, 2017.
[23] Y. Zhang, A. Afanasyev, J. Burke, L. Zhang, A survey of mobility support in named data networking, in: 2016 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), IEEE, 2016, pp. 83–88.
[24] S. Shannigrahi, C. Fan, C. Partridge, What's in a name? naming big science data in named data networking, in: Proceedings of the 7th ACM Conference on Information-Centric Networking, 2020, pp. 12–23.
[25] T. Nguyen, X. Marchal, G. Doyen, T. Cholez, R. Cogranne, Content poisoning in named data networking: Comprehensive characterization of real deployment, in: 2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM), 2017.
[26] L. Wang, A. Hoque, C. Yi, A. Alyyan, B. Zhang, Ospfn: An ospf based routing protocol for named data networking, Tech. rep., Technical Report NDN-0003 (2012).
[27] H. Dai, J. Lu, Y. Wang, B. Liu, A two-layer intra-domain routing scheme for named data networking, in: 2012 IEEE Global Communications Conference (GLOBECOM), IEEE, 2012.
[28] A. M. Hoque, S. O. Amin, A. Alyyan, B. Zhang, L. Zhang, L. Wang, Nlsr: Named-data link state routing protocol, in: Proceedings of the 3rd ACM SIGCOMM workshop on Information-centric networking, 2013.
[29] A. Afanasyev, J. Burke, T. Refaei, L. Wang, B. Zhang, L. Zhang, A brief introduction to named data networking, in: MILCOM 2018-2018 IEEE Military Communications Conference (MILCOM), IEEE, 2018, pp. 1–6.
[30] G. White, G. Rutz, Content delivery with content-centric networking, CableLabs, Strategy & Innovation (2016) 1–26.
[31] S. Mastorakis, A. Afanasyev, Y. Yu, L. Zhang, ntorrent: Peer-to-peer file sharing in named data networking, in: 2017 26th International Conference on Computer Communication and Networks (ICCCN), IEEE, 2017, pp. 1–10.
[32] M. Hossain, Y. Karim, R. Hasan, Secupan: A security scheme to mitigate fragmentation-based network attacks in 6lowpan, in: Proceedings of the Eighth ACM Conference on Data and Application Security and Privacy, 2018, pp. 307–318.
[33] G. Montenegro, N. Kushalnagar, J. Hui, D. Culler, et al., Transmission of ipv6 packets over ieee 802.15. 4 networks, Internet proposed standard RFC 4944 (2007) 130.
[34] Z. Zhang, E. Lu, Y. Li, L. Zhang, T. Yu, D. Pesavento, J. Shi, L. Benmohamed, Ndnot: a framework for named data network of things, in: Proceedings of the 5th ACM Conference on Information-Centric Networking, 2018, pp. 200–201.
[35] Y. Zhang, D. Raychadhuri, R. Ravindran, G. Wang, Icn based architecture for iot, IRTF contribution, October (2013).
[36] M. A. Hail, I. Pösse, S. Fischer, Integration of fiware and iot based named data networking (iot-ndn)., in: SENSORNETS, 2022, pp. 184–190.
[37] L. Pi, L. Wang, Secure bootstrapping and access control in ndn-based smart home systems, in: IEEE INFOCOM 2018-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), IEEE, 2018, pp. 1–2.
[38] S. H. Ahmed, D. Kim, Named data networking-based smart home, Ict Express 2 (3) (2016) 130–134.
[39] Z. Zhang, T. Yu, X. Ma, Y. Guan, P. Moll, L. Zhang, Sovereign: Self-contained smart home with data-centric network and security, IEEE Internet of Things Journal (2022).
[40] A. A. Ramadha, L. V. Yovita, T. A. Wibowo, Design and implementation named data networking-based video streaming system, in: 2022 5th International Conference on Information and Communications Technology (ICOIACT), IEEE, 2022, pp. 66–70.
[41] J. Burke, Video streaming over named data networking, E-LETTER (2013).
[42] H. Li, Y. Li, T. Lin, Z. Zhao, H. Tang, X. Zhang, Merts: A more efficient real-time traffic support scheme for content centric networking, in: 6th International Conference on Computer Sciences and Convergence Information Technology (ICCIT), IEEE, 2011.
[43] C. A. Kerrche, F. Ahmad, M. Elhoseny, A. Adnane, Z. Ahmad, B. Nour, Internet of vehicles over named data networking: current status and future challenges, in: Emerging Technologies for Connected Internet of Vehicles and Intelligent Transportation System Networks, Springer, 2020, pp. 83–99.
[44] Z. Sabir, A. Amine, Ndn vs tcp/ip: Which one is the best suitable for connected vehicles?, in: Recent Advances in Mathematics and Technology, Springer, 2020, pp. 151–159.
[45] E. Barka, C. A. Kerrache, R. Hussain, N. Lagraa, A. Lakas, S. H. Bouk, A trusted lightweight communication strategy for flying named data networking, Sensors (2018) 2683.
[46] M. Amadeo, C. Campolo, A. Molinaro, Crown: Content-centric networking in vehicular ad hoc networks, IEEE Communications Letters 16 (9) (2012) 1380–1383.
[47] M. Chen, D. O. Mau, Y. Zhang, T. Taleb, V. C. Leung, Vendnet: Vehicular named data network, Vehicular Communications 1 (4) (2014) 208–213.
[48] H. Khelifi, S. Luo, B. Nour, A. Sellami, H. Moungla, F. Naït-Abdesselam, An optimized proactive caching scheme based on mobility prediction for vehicular networks, in: 2018 IEEE global communications conference (GLOBECOM), IEEE, 2018, pp. 1–6.
[49] A. Aboud, H. Touati, B. Hnich, Hybrid 802.11 p-cellular architecture for ndn-based vanet, International Journal of Communication Systems 36 (3) (2023) e5393.
[50] R. Ullah, M. A. U. Rehman, B.-S. Kim, Design and implementation of an open source framework and prototype for named data networking-based edge cloud computing system, IEEE Access 7 (2019) 57741–57759.
[51] W. Parker, Climate Science, in: E. N. Zalta (Ed.), The Stanford Encyclopedia of Philosophy, Summer 2018 Edition, Metaphysics Research Lab, Stanford University, 2018.
[52] D. H. Perkins, D. H. Perkins, Introduction to high energy physics, CAMBRIDGE university press, 2000.
[53] S. Shannigrahi, C. Fan, C. Papadopoulos, Request aggregation, caching, and forwarding strategies for improving large climate data distribution with ndn: a case study, in: Proceedings of the 4th ACM Conference on Information-Centric Networking, 2017.
[54] D. Saxena, V. Raychoudhury, N. SriMahathi, Smarthealth-ndnot: Named data network of things for healthcare services., in: MobileHealth@ MobiHoc, 2015, pp. 45–50.
[55] D. Saxena, V. Raychoudhury, Design and verification of an ndn-based safety-critical application: A case study with smart healthcare, ieee transactions on systems, man, and cybernetics: systems 49 (5) (2017).
[56] D. N. Kanellopoulos, Congestion control for ndn-based manets: Recent advances, enabling technologies, and open challenges, Journal of Organizational and End User Computing (JOEUC) 33 (5) (2021) 111–134.
[57] D. Gupta, S. Rani, S. Raza, N. M. F. Qureshi, R. F. Mansour, M. Ragab, Security paradigm for remote health monitoring edge devices in internet of things, Journal of King Saud University-Computer and Information Sciences (2023) 101478.
[58] Z. Zhang, A. Afanasyev, L. Zhang, Ndncert: universal usable trust management for ndn, in: Proceedings of the 4th ACM Conference on Information-Centric Networking, 2017, pp. 178–179.
[59] Y. Yu, Usable security for named data networking, University of California, Los Angeles, 2016.
[60] C. Ghali, G. Tsudik, E. Uzun, Network-layer trust in named-data networking, ACM SIGCOMM Computer Communication Review 44 (5) (2014) 12–19.
[61] L. Zhang, D. Estrin, J. Burke, V. Jacobson, J. D. Thornton, E. Uzun, B. Zhang, G. Tsudik, kc claffy, D. Krioukov, D. Massey, C. Papadopou-

los, P. Ohm, T. Abdelzaher, K. Shilton, L. Wang, E. Yeh, P. Crowley, Ndn packet format specification 0.3: Signature, Named Data Networking (2022).
[62] A. Rowstron, G. Pau, Characteristics of a vehicular network, University of California Los Angeles, Computer Science Department, Tech. Rep (2009).
[63] S. K. Ramani, A. Afanasyev, Rapid establishment of transient trust for ndn-based vehicular networks, in: 2020 IEEE International Conference on Communications Workshops (ICC Workshops), IEEE, 2020, pp. 1–6.
[64] P. Gasti, G. Tsudik, E. Uzun, L. Zhang, Dos and ddos in named data networking, in: 2013 22nd International Conference on Computer Communication and Networks (ICCCN), IEEE, 2013.
[65] D. Wu, Z. Xu, B. Chen, Y. Zhang, What if routers are malicious? mitigating content poisoning attack in ndn, in: 2016 IEEE Trustcom/BigDataSE/ISPA, IEEE, 2016, pp. 481–488.
[66] A. Compagno, M. Conti, E. Losiouk, G. Tsudik, S. Valle, A proactive cache privacy attack on ndn, in: NOMS 2020-2020 IEEE/IFIP Network Operations and Management Symposium, 2020.
[67] Y. Yu, Y. Li, X. Du, R. Chen, B. Yang, Content protection in named data networking: Challenges and potential solutions, IEEE Communications Magazine 56 (11) (2018) 82–87.
[68] What is a denial of service attack?, Paloalto Networks `https://www.paloaltonetworks.com/cyberpedia/what-is-a-denial-of-service-attack-dos` (2020).
[69] S. K. Ramani, R. Tourani, G. Torres, S. Misra, A. Afanasyev, Ndn-abs: Attribute-based signature scheme for named data networking, in: Proceedings of the 6th ACM Conference on Information-Centric Networking, 2019, pp. 123–133.
[70] M. Yu, R. Li, Y. Liu, Y. Li, A caching strategy based on content popularity and router level for ndn, in: 2017 7th IEEE International Conference on Electronics Information and Emergency Communication (ICEIEC), IEEE, 2017, pp. 195–198.
[71] J. Hou, H. Lu, A. Nayak, A gnn-based proactive caching strategy in ndn networks, Peer-to-Peer Networking and Applications 16 (2) (2023) 997–1009.
[72] H. Im, D. Kim, An overview of content poisoning in ndn: Attacks, countermeasures, and direction, KSII Transactions on Internet and Information Systems (TIIS) 14 (7) (2020).
[73] X. Hu, J. Gong, G. Cheng, G. Zhang, C. Fan, Mitigating content poisoning with name-key based forwarding and multipath forwarding based inband probe for energy management in smart cities, IEEE Access 6 (2018) 39692–39704.
[74] L. Zhang, D. Estrin, J. Burke, V. Jacobson, J. D. Thornton, D. K. Smetters, B. Zhang, G. Tsudik, D. Massey, C. Papadopoulos, et al., Named data networking (ndn) project, Relatório Técnico NDN-0001, Xerox Palo Alto Research Center-PARC 157 (2010) 158.
[75] Y. Wang, H. Dai, J. Jiang, K. He, W. Meng, B. Liu, Parallel name lookup for named data networking, in: 2011 IEEE Global Telecommunications Conference-GLOBECOM 2011, IEEE, 2011.
[76] Y. Wang, K. He, H. Dai, W. Meng, J. Jiang, B. Liu, Y. Chen, Scalable name lookup in ndn using effective name component encoding, in: 2012 IEEE 32nd International Conference on Distributed Computing Systems.
[77] Y. Wang, Y. Zu, T. Zhang, K. Peng, Q. Dong, B. Liu, W. Meng, H. Dai, X. Tian, Z. Xu, et al., Wire speed name lookup: A gpu-based approach, in: 10th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 13), 2013.
[78] S. DiBenedetto, P. Gasti, G. Tsudik, E. Uzun, Andana: Anonymous named data networking application, arXiv preprint arXiv:1112.2205 (2011).
[79] R. Kaur, A. Singh, A. Singh, A. Goyal, A. Singh, S. Batra, An efficient pending interest table content search in ndn through stable bloom filter, The Computer Journal (2023) bxad033.
[80] M. Tortelli, L. A. Grieco, G. Boggia, K. Pentikousisy, Cobra: Lean intradomain routing in ndn, in: 2014 IEEE 11th Consumer Communications and Networking Conference (CCNC), IEEE, 2014.
[81] F. Berto, L. Calderoni, M. Conti, E. Losiouk, Spatial bloom filter in named data networking: a memory efficient solution, in: Proceedings of the 35th Annual ACM Symposium on Applied Computing, 2020.
[82] T. Mick, R. Tourani, S. Misra, Laser: Lightweight authentication and secured routing for ndn iot in smart cities, IEEE Internet of Things Journal 5 (2) (2017) 755–764.
[83] N. Yang, K. Chen, M. Wang, Smartdetour: Defending blackhole and content poisoning attacks in iot ndn networks, IEEE Internet of Things Journal 8 (15) (2021) 12119–12136.
[84] Q. Li, X. Zhang, Q. Zheng, R. Sandhu, X. Fu, Live: Lightweight integrity verification and content access control for named data networking, IEEE Transactions on Information Forensics and Security 10 (2) (2014).
[85] H. Huang, Y. Wu, F. Xiao, R. Malekian, An efficient signature scheme based on mobile edge computing in the ndn-iot environment, IEEE Transactions on Computational Social Systems (2021).
[86] J. Lou, Q. Zhang, Z. Qi, K. Lei, A blockchain-based key management scheme for named data networking, in: 2018 1st IEEE International Conference on Hot Information-Centric Networking (HotICN), IEEE, 2018, pp. 141–146.
[87] R. Chatterjee, S. Ruj, S. DasBit, Public key infrastructure for named data networks, in: Proceedings of the 21st International Conference on Distributed Computing and Networking, 2020.
[88] A. Albalawi, J. Garcia-Luna-Aceves, Named-data transport: An end-to-end approach for an information-centric ip internet, in: Proceedings of the 7th ACM Conference on Information-Centric Networking, 2020.
[89] S. Shannigrahi, C. Fan, G. White, Bridging the icn deployment gap with ipoc: An ip-over-icn protocol for 5g networks, in: Proceedings of the 2018 Workshop on Networking for Emerging Applications and Technologies, 2018, pp. 1–7.
[90] H. Wu, J. Shi, Y. Wang, Y. Wang, G. Zhang, Y. Wang, B. Liu, B. Zhang, On incremental deployment of named data networking in local area networks, in: 2017 ACM/IEEE Symposium on Architectures for Networking and Communications Systems (ANCS), IEEE, 2017, pp. 82–94.
[91] Z. Lin, M. Kuai, X. Hong, Reliable forwarding strategy in vehicular networks using ndn, in: 2016 IEEE 84th Vehicular Technology Conference (VTC-Fall), IEEE, 2016, pp. 1–5.
[92] M. Burhan, R. A. Rehman, Bsms: a reliable interest forwarding protocol for ndn based vanets, in: 2020 3rd International Conference on Advancements in Computational Sciences (ICACS), IEEE, 2020, pp. 1–6.
[93] Z. Rezaeifar, J. Wang, H. Oh, S.-B. Lee, J. Hur, A reliable adaptive forwarding approach in named data networking, Future Generation Computer Systems 96 (2019) 538–551.
[94] J. Lai, L. Tian, W. Ma, J. Zhu, Enhanced transmission control strategies for reliable live streaming in ndn-based uav swarm networks, in: 2022 IEEE 5th International Conference on Electronics Technology (ICET), IEEE, 2022, pp. 606–611.
[95] M. Zhang, Y. Chen, Inductive matrix completion based on graph neural networks, arXiv preprint arXiv:1904.12058 (2019).
[96] L. Berger, I. Bryskin, A. Zinin, R. Coltun, The ospf opaque lsa option, Tech. rep., RFC 5250, July (2008).
[97] R. C. Merkle, A digital signature based on a conventional encryption function, in: Conference on the theory and application of cryptographic techniques, Springer, 1987, pp. 369–378.
[98] G. Wood, et al., Ethereum: A secure decentralised generalised transaction ledger, Ethereum project yellow paper 151 (2014) (2014) 1–32.
[99] M. Conti, A. Gangwal, M. Hassan, C. Lal, E. Losiouk, The road ahead for networking: A survey on icn-ip coexistence solutions, IEEE Communications Surveys & Tutorials (2020).
[100] R. Zhu, T. Li, T. Song, igate: Ndn gateway for tunneling over ip world, in: 2021 International Conference on Computer Communications and Networks (ICCCN), IEEE, 2021, pp. 1–9.
[101] A.-q. Majed, X. Wang, B. Yi, Name lookup in named data networking: A review, Information 10 (3) (2019) 85.
[102] A. Anjum, S. Hounsinou, H. Olufowobi, Work-in-progress: Deadline-aware named data networking for time-sensitive iot applications, in: Proceedings of the 29th IEEE Real-Time and Embedded Technology and Applications Symposium, 2023 (in-press).
[103] A. Anjum, H. Olufowobi, Towards mitigating blackhole attack in ndn-enabled iot, in: 2023 IEEE International Conference on Consumer Electronics (ICCE), IEEE, 2023, pp. 1–6.