A Survey on Information-Centric Networking: Rationales, Designs and Debates

JIANG Xiaoke[1,2,3], BI Jun[1,2,3], NAN Guoshun[4], LI Zhaogeng[1,2,3]

[1] Institute for Network Sciences and Cyberspace, Tsinghua University
[2] Department of Computer Science and Technology, Tsinghua University
[3] Tsinghua National Laboratory for Information Science and Technology
[4] State Key Laboratory of Networking and Switching Technology, BUPT

Abstract: The basic function of the Internet is to delivery data (what) to serve the needs of all applications. IP names the attachment points (where) to facilitate ubiquitous interconnectivity as the current way to deliver data. The fundamental mismatch between data delivery and naming attachment points leads to a lot of challenges, e.g., mapping from data name to IP address, handling dynamics of underlying topology, scaling up the data distribution, and securing communication, etc. Information-centric networking (ICN) is proposed to shift the focus of communication paradigm from where to what, by making the named data the first-class citizen in the network, The basic consensus of ICN is to name the data independent from its container (space dimension) and session (time dimension), which breaks the limitation of point-to-point IP semantic. It scales up data distribution by utilizing available resources, and facilitates communication to fit diverse connectivity and heterogeneous networks. However, there are only a few consensuses on the detailed design of ICN, and quite a few different ICN architectures are proposed. This paper reveals the rationales of ICN from the perspective of the Internet evolution, surveys different design choices, and discusses on two debatable topics in ICN, i.e., self-certifying versus hierarchical names, and edge versus pervasive caching. We hope this survey helps clarify some mis-understandings on ICN and achieve more consensuses.

Keywords: information-centric networking, content-centric networking, future internet architecture, named-data networking, publish-subscribe

I. INTRODUCTION

The Internet design is dated back to 1960s and ‘70s when resource sharing was the primary goal. Some scarce and expensive devices like card readers, high-speed tape drives and computers were hosted on the limited number of sites, and shared by the community. Internet Protocol (IP)[1], which names the attachment point, was designed to support the resource sharing in the mainframe era. However, the Internet has evolved to a very different one from what it was. On one side, according to Cisco’s VNI report[2], the compound annual growth rate of network traffic is anticipated to be 29% during 2011-2016, and traffic alone will account for 86% of all the traffic in 2016. Some powerful cloud/service platforms are built for the large scale data distribution. On the other side, resources (electronic content, computing, storage, etc) are not only spread to client sides connected with stationary attachment point as in PC era, but also distributed everywhere, including different devices with diverse connectivity or even in heterogeneous networks, such as mobile phones, wearable devices, sensors, vehicles, and satellites. IP, which is now underpinning such an cyberspace, has indeed exceeded designer's expectations. After all, even the designers of the Internet had not envisioned the myriad ways in which it is used today.

However, end users essentially care about "what" rather than "where", and the basic function of the Internet is to ship data to serve the needs of applications. When IP, focusing on talking to whom (where), is applied to deliver data (what) in current era, the fundamental mismatch leads to a lot of issues, including but not limited to the followings:

- Data is forwarded following the specific path defined by the routing spanning tree, without the ability to utilize extra interfaces, multiple end-to-end paths or redundant data replicas. This prevents data providers from scaling up the data distribution.
- IP, as originally designed, provides no security support. Solutions were added later to secure the session, e.g., TLS, IPSec. However, these solutions only provide transient trust – the trust is valid for the two endpoints exclusively (space constraint) during the session period only (time constraint).
- Naming the attachment point leads to binding with underlying identifier (layer-2 identifier), which makes it hard to support multi-homing and handle dynamic change of underlying topology, such as mobility support, ad-hoc scenarios. This is especially true for some heterogeneous networks, e.g., sensors network, vehicle network, delay-torrent network (DTN), wherein it is rarely possible to build steady end-to-end channel to ship bits.
- IP is ill-fited for some new applications where location or even device identity is not important, such as Internet of Thing (IoT), sensor network, vehicle network. For those applications, it is the data that interests people, instead of where the location is, or which device provides the data. For example, a driver may be interested in the distance (data) from his/her car to the nearby cars instead of those car themselves, a housekeeper may want to know the temperature (data) in a house without caring which sensor provides that data if there are multiple.

A clean-slate method to address the above issues is abandoning the IP paradigm which was designed for resource sharing on the limited number of sites 50+ years ago, replacing *where* with *what* to ship bits. Hence, Information-centric Networking (ICN) is proposed by naming the data directly, which breaks the limitation of point-to-point IP semantic. ICN scales up the data distribution by utilizing redundant resources, and facilitates secure, efficient and flexible data delivery to fit diverse connectivity and heterogeneous networks.

Naming the data is fundamental idea of ICN, not complete design of an architecture. How to create a network architecture based on named data? An ICN architecture, besides naming the data, should support another two functions: 1) retrieving target data, and 2) securing the data.

Extensive studies have been done in ICN research, and quite a few ICN achitectures have been proposed so far due to lack of consensuses on the design. Here we list some of these proposals: Data-Oriented Network Architecture (DONA)[3], Publish Subscribe Internet Technology (PURSUIT)[4], and its predecessor Publish-Subscribe Internet Routing Paradigm (PSIRP)[5], Network of Information (NetInf)[6], [7], which was initially conceived as Architecture and Design for the Future Internet (4WARD)[8], and evolved further as Scalable and Adaptive Internet Solutions (SAIL)[9], Named-Data networking (NDN) [10], which has its roots from an earlier project, Content-Centric Networking (CCN)[11]. More proposals and detailed design survey can be found in the existing literatures[12], [13], [14], [15], [16].

After briefly intruding and comparing different design choices adopted by major ICN proposals, this paper presents the most debatable topics in ICN, including naming structure and caching location.

In this survey, we give an insight into the ICN and its design, instead of the design details or recognized challenges in different proposals. The rest of this paper is structured as follows. In Section II, we explain the necessity of ICN network architecture, together with efforts (and failures) of improving data delivery over IP. Section III analyzes the required functions of ICN (naming, retrieving and securing the data) as well as the possible design choices from the perspective of network architecture. Section IV presents debates on the fundamental design, i.e., self-certifying versus hierarchical names, edge versus pervasive caching, which are hot topics in ICN research community. Finally, we summarize the survey in Section V.

II. Motivations of ICN

IP is being used for data delivery. To overcome the issues caused by IP's shortcomings as mentioned in preceding section, researchers have invented many fixes.

- Large scale data distribution is being overlayed on top of the underlying IP network topology. End users request data by name, e.g, URL, instead of connecting given devices first. The data is returned from the device picked by the overlay infrastructure instead of the one chosen by end users. Domain names are merely aliases of containers, but refer to data. This is especially true in content delivery network (CDN), and peer-to-peer (p2p) system.
- Heterogeneous networks, such as sensor network, vehicle network, and satellite network, are isolated from the Internet. Gateway is usually required to ensure the communication between two heterogeneous network. Furthermore, dedicated protocols are created for heterogeneous networks.

What is more, a lot of works have been done to improve the performance of the data delivery over IP, and these studies can be divided into patch and overlay fashion.

Some researchers developed incremental solutions l to better support data distribution, e.g., IP multicast[17], multipath TCP[18], multipath routing[19], stream control transmission protocol (SCTP)[20], Datagram Congestion Control Protocol (DCCP)[21], and transport next-generation (Tng)[22]. All of these works try to break through the naming semantics of IP, and use the IP namespace (together with port and sequence number) to identify something else, e.g., communication groups, point-to-point paths, or message-oriented data chunk. However, these piecemeal solutions have not been deployed globally due to some critical reasons, like inter-working with the existing IP systems, lack of evident business incentive, etc. On the other hand, overlay systems are widely used to improve the performance over IP, e.g., CDN, p2p, application-layer multicast (ALM)[23]. However, the overlay systems are far from perfection due to the mismatch between the application-layer goals and point-to-point semantic of underlying protocols. Overlay systems suffer from the trust in underlying networks, handling heterogeneity of users, providing resilience, higher path stretch and a high link stress, etc. For example, CDN has to fool the end users that they are talking with the intended entity in some hack way, while the truth is the surrogates that they are connecting to, are owned by CDN providers. The third-party role of surrogates becomes fundamental conflict of trust management, and leads to potential risks for HTTP Secure (HTTPS) based communication [24]. P2p file-sharing system relies on application-layer routing, which is largely independent from the Internet routing and topology. Thus, it leads to a lot of unwanted inter-domain traffic, and starves other applications, such as web traffic[25].

A fundamental issue of patches and overlay systems is that, each of them is a piecemeal solution of one specific problem. The inefficiency of point-to-point communication, failure of piecemeal solutions, motivate researchers to rethink the Internet architecture in a "clean-slate" way. A basic observation is that end users essentially concern the data they desired without caring where the data locates, how the data is reached, or from which path the data is transmitted. Hence, Information-centric networking (ICN) is proposed as a revolutionary architecture, which grants the data the first-class citizen in the network by naming the data directly. ICN is not a patch of IP or overlay over IP, but a brand new network paradigm for future Internet.[1]

By naming the data, ICN supports data retrieval with given data names. To scale up data distribution, resources, such as multiple data replicas (authoritative sources or the delegation), multiple local link interfaces (e.g., ethernet, WiFi, 2/3/4G, bluetooth, infrared radiation), and multiple physical connected paths between two ends can be supported naturally. While techniques, such as multicast, identical traffic aggregation onrouter path selection and traffic control, broadcast especially on broadcast channel can be implemented seamlessly to reduce traffic or handle dynamic of traffic and underlying topology.

Moreover, since the named data in ICN is decoupled from the container and the session, ICN facilitates users' mobility without reestablishment of sessions, as well as data communication between heterogeneous networks wherein frequent access points handover or topology change is frequent. For those applications with data-centric requirement (e.g., IoT, vehicle network), ICN is the best fitted underlying protocol. Every piece of data is named, which then can be flooded via broadcast channel. Thus, whoever is interested in the data just accept what his/she desires with little cost.

End users are expected to verify the received data before they really consume the encapsulated data in order to ensure the data is the desired one, from the intended source, and unmodified in transit. To decouple the security of data from the containers and sessions, ICN secures data directly by including security information to data itself, which is call *data-oriented security*. Data-oriented security allows the data to be verified independent from where it is stored by whoever retrieves it.

Studies showed that in-network caching can significantly improve the network performance, in terms of increasing the throughput, reducing the network traffic and retrieval delay, and saving the congestion collapse. Naming the data in network layer makes in-network caching intrinsically supported, while the rationale to trade storage for bandwidth underneath is the business consideration: the cost of storage has been decreasing faster than the cost of bandwidth, and it is expected that the storage capacity that exists in most networked devices (either deployed in the fixed infrastructure or carried by endusers) will greatly expand in the near future. Therefore, it is widely accepted as an enhancement in ICN. It is worth note that in-network caching is an important design component, but far from the all of ICN. As we can see from the whole picture of ICN, ICN is a fundamental change to the Internet architecture. Hence, it cannot be interpreted to enhance current architecture with caching support.

III. Overview of ICN: Concepts, Terminologies, and Designs

Since different ICN proposals advocate different terminologies to represent the design elements, here we unify those terminologies for common elements. The basic data units that are transmitted over ICN is called Named Data Objects (NDOs). Note that NDOs are in different granularity in different proposals, which can be webpages, videos, documents, or data packets. Usually, a NDO contains a name which is unique within the defined scope, the bits that represent electronic data of application, a digital signature that binds its name and data, and the public key or its pointer whose associated private key is used to generate the signature. A NDO is the response to a *request* with the matched name.

There are three basic roles in ICN network: 1) the end applications that originate NDOs are called producers; producers are equipped with private-public keys, and should publish some accessing information in the network to make its data available to the public; 2) the

[1] At preliminary stage, overlay deployment is main approach adopted by ICN, and some ICN proposals do pursue the compatibility with IP.

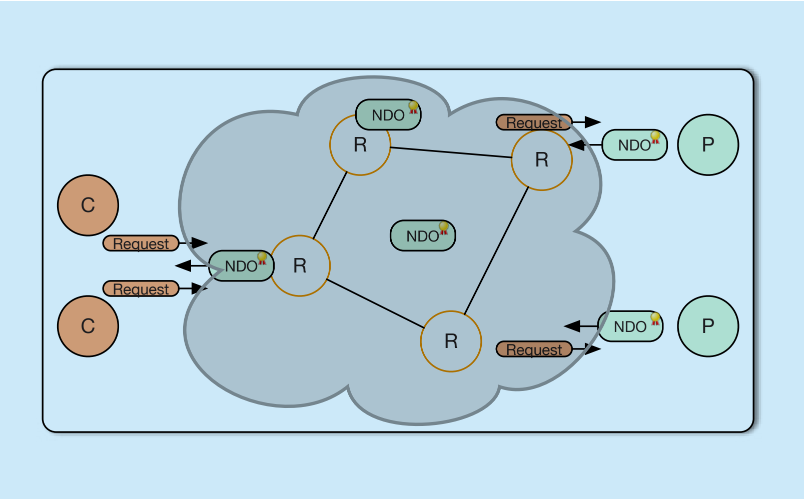

Fig.1 *ICN Framework Overview. The system includes 1) three roles, i.e., consumer (C), producer (C) and router (R); 2) two kinds of messages, i.e., request and NDO, note that badges on the upper right corner of NDO indicates signature. Session is unnecessary for request/NDO exchange. NDO can be verified wherever it is stored and by whoever retrieves it*

Table I *Design summary*

Feature	DONA	NDN	PURSUIT	SAIL
NDO Naming	self-certifying	hierarchical	self-certifying	hierarchical or self-certifying
Routing Identifier	attachment point address	data name	path label	attachment point address
Data Discovery	name-based routing	name-based routing	name-based routing	name-based routing/ (name resolution & locator-based routing)
Data Delivery	locator-based routing / hop-by-hop state	hop-by-hop state	source routing	locator-based routing / hop-by-hop state

end applications that originate initial requests are called consumers; consumers retrieve NDOs by sending out requests to the network; and 3) the network equipments that forward packets are called content routers (CRs). Any device may play one or two or all three roles at the same time.

An overview of ICN framework is presented in Figure 1, wherein named data, session-less request/NDO exchange, data-oriented security, and in-network caching are highlighted. In the rest of this section, we summarize the design choices of the basic three functions of ICN, i.e., naming, retrieving and securing the data. We give a design summary of some representative ICN proposals in Table I: NDN adopts hierarchical name and name-based routing; DONA adopts self-certifying name; PURSUIT adopts path addressing to forward NDO; and SAIL allows hybrid way.

3.1 Naming the NDO

Naming is the first and foremost element of ICN, since the ICN primitives are based on the NDO names.

To identify a piece of data independent from where it is stored, how it is reached, or which path it is transmitted, the NDO names must be 1) unique to identify different NDOs, and 2) persistent to preserve its uniqueness independent from the container (space dimension) and session (time dimension). There are mainly two kinds of naming schemes are proposed: hierarchical and self-certifying names.

Hierarchical name follows the naturally existing naming schemes used by millions of applications. It is constructed by multiple components with logical hierarchy, and some components are naturally human-readable. Domain name is such an example, e.g., “www.google.com/maps/@44.8,-100.9,5z”. Hierarchical name is what the end users and applications use to retrieve data. Hierarchical name is also the basis to secure data, since it is the end users who confirm whether a received data is the desired one, or produced by the trusted producer within the context of given by hierarchical name(Section III-C).

Self-certifying name is necessarily non-structured, i.e., flat, or concatenation of multiple flat components. It is cryptographically constructed so that one can securely determine whether a given piece of data matches a given name. That is why it is called “self-certifying” name.[2] A simple form of self-certifying naming is to simply name a piece of data directly by its cryptographic (e.g., SHA-1) digest, which is widely adopted in p2p system. A general form of self-certifying name is P:L, where P is a cryptographic digest of the producer’s public key and L is the label to make the content unique.[3] The use of cryptographic hashing function for computing P provides the binding[4] between the name and the key, by enabling the receiver to check that the key indeed hashed to P. That is, if someone claims that a key is associated with a name P, we can simply compute the hash to confirm it, although the existence of this binding does not mean that knowing P is enough to derive the key. The main advantage of self-certifying name is to prevent cache poisoning by verifying the NDOs directly without further dependence[5].

Both two naming schemes can identify a piece of data independent from where it is stored, how it is reached, or which path it is transmitted. But their difference is also obvious.

Hierarchical names follows the existing naming convention in human's mind, which is also applied to man-developed software applications. Thus, hierarchical name provides usability and trust. Even though consumers can only remember or identify a very small set of hiearchical names, those names help a lot to retrieve and secure the data. some names are the door of the whole cyberspace (but may be not trust source). For example, "www.google.com" which links the Google search engine, and then can provide names of almost any other NDOs with human's selection. Some names serve as trusted source of names of more NDOs, e.g., "www.chase.com" contains a lot of links (names) of functionality provided by this bank, e.g., log in, money transfer. In this case, even if end users cannot recognize the names of linked NDOs, especially when it contains non-human-readable component, they can still trust that the names link to desired data, since the names come from the trusted source.

Self-certifying name is purely concept in cyberspace, which is strongly coupled with content validation. But it possesses little of usability and trust, and requires further assumption/dependency. For example, p2p file-sharing system adopts self-certifying name. It usually needs a magic file to store names of target NDOs, i.e.,torrent file for BitTorrent, ed2k file for eDonkey. To download a desired data, the p2p users must 1) securely retrieve the magic file in some magic way, and 2) what is more essential, believe the magic file truly contains the name of their desired data.

We present detailed comparison on hierarchical v.s. selfcertifying names in Section IV-A.

3.2 Retrieving the NDO

The retrieval of NDO can be divided into two steps. The first step is the data discovery: requests are forwarded towards where data may be located. The second step is the data delivery that NDOs are transmitted to the consumers. What is more, the NDOs can also be cached along the very path to satisfy future requests.

1) Data Discovery: For data discovery, some ICN proposals employ a straightforward routing scheme: producers first announce data names or name prefixes to the routing system, so that the CRs can gather the announcements and update routing tables for the subsequent requests forwarding (NDN). This is the so-called "name-based routing". Some ICN proposals achieve name-based routing with an additional systems, instead of CRs.[6] For example, Resolution Handler in DONA and rendezvous point in PURSUIT can be viewed as this kind of system. Some other ICN proposals introduce an extra routing identifier beyond name. The most common approach is to map a name to the location of the producer before forwarding requests with the resolved locator (SAIL). This approach is similar to current HTTP over IP, wherein a URL (NDO name) is first mapped to an IP address of web server (producer) by DNS/DNSSEC, and then the request is forwarded based on this IP address.

Note that one can easily retrieve local data without announcement by utilizing broadcast channel.

2) Data Delivery: For data delivery, some ICN proposals deliver the NDOs following the exact reverse path of the corresponding requests (NDN). This is implemented with hop-by-hop *soft state*, indicating that each CR along the request path creates temporary records (incoming and outgoing interfaces,

[2] Self-certifying names are also used by some applications, but it usually needs to be mapped from hierarchical name.

[3] Self-certifying name cannot bind the P with the content, thus it also need extra signature to bind the name with content[26], unless the L is the signature itself.

[4] It is more the same entity in two formats, rather than a "binding".

[5] Given the public key is provided.

[6] For these solutions, they also need routers to forward packets based on different routing identifier, e.g., IP and path label, as shown in Table I

the request, etc.) for the forwarded requests. Thus, NDOs can be returned to the consumers by checking the incoming interfaces hop-by-hop. The value of hop-by-hop state is beyond data delivery. It builds a symmetric path for the request/NDO exchange, facilitates the aggregation of identical requests (i.e., built-in support for multicast), enables hop-by-hop traffic control, and also eliminate packet loop. So that CRs can freely explore multiple paths in retrieving data.

Some other ICN proposals leverage an additional routing identifier for data delivery, such as consumer locator (DONA, SAIL) with HTTP over IP mechanism, and delivery path identifier with mechanism similar to source routing (PURSUIT).

3) Caching: Caching is an important feature of ICN architectures. By naming the data in network layer, ICN can employ transparent and universal in-network caching for efficient data distribution. Further, unlike the application dependent caching in TCP/IP networks, caches in ICN are expected to serve different traffics generated by applications such as videos and web, since the NDOs are detached from applications and host related information.

Recent research for ICN caching can be mainly divided into two directions. The first direction is about cache replacement policies, and the second is about caching storage placement strategies. [27] studied the impact of cache replacement policies (e.g., LRU, LFU, FIFO) to the service improvement and found that the simple LRU is the best candidate. There are also some research on on-demand caching, e.g., popularity-based caching[28], age-based cooperative caching[29]. As to cache storage placement strategies, [30] argues that routers at the edge contribute the decisive portion of network traffic reduction based on dataset from a CDN provider. While [31] implies the pervasive caching is the better strategies based on data collected from the ISP. And this has been an impressing debatable topic as discussed in Section IV-B. Since caching is not the main focus of this survey, we only mention a few sample work in the above. Survey in [32], [33], [34] present more details of current caching research.

3.3 Securing the NDO

By including a signature to NDO directly, ICN allows the data to be verified wherever it is stored and whoever retrieves it. The security is based on data itself, and decoupled from container and session. The following three attributes of NDO should be verified to secure the data[26]: 1) validity (including integrity and authenticity in traditional notions), that the received NDO is a complete, uncorrupted copy of what the publisher sent; 2) provenance, that the producer is a trusted one to supply this NDO; and 3) relevance, that the received NDO is the desired one.

[26] proposes three steps to secure a network content: 1) verifying that a given name-content mapping was signed by a particular key; 2) determining something about whom that key belongs to, in our term, the producer; and 3) deciding whether or not that is an acceptable producer for this particular data and the use to which it is to be put. In above terminology, the first step determines validity, the latter two steps determine provenance, and the name itself, along with the means by which it was obtained and the third step above, determine relevance.

[35] proposes availability as a goal of securing the network, which mainly refers to protect the network against content-level denial-of-service caused by cache poisoning.

The four attributes, relevance, provenance, validity and availability are in different level.

Validation of validity is purely syntactic: it simply verifies that NDO is signed by the key it purports (the key whose fingerprint is specified as the content publisher) based on specialized mechanical operation. Provenance and relevance are agreements between producers and consumers, and what is appropriate for one application might not be appropriate for another. While availability is a capability of routers rather than an attribute of NDOs like the other three. And the way to achieve availability is to enable routers to verify the NDOs (validity, provenance and relevance).

The validation of provenance and relevance should be syntactic after manually and/or automatic configuration, e.g., choosing the accepted trust anchors and trust policies. By this means, validation of validity, provenance and relevance are syntactic. Thus, routers can automate NDOs verification to achieve availability. The only obstacle is to spread the accepted trust anchors and trust policies to routers.

Self-certifying name simplifies the verification by prior provenance and relevance: the desired electronic content can be mapped to label L with cryptographic way with given public key P. Thus, availability under self-certifying name becomes validity.

Under the same assumption, i.e., prior provenance and relevance, hierarchical name achieve the same simplification as self-certifying name, which is referred as *evidence-based security*[11], [26]. For example, if the digest of target NDO is pre-known and trusted by the consumers, hierarchical name can contain the digest as the last component. In this case, NDOs verification is simplified to validate the digest.

As for general situation for hierarchical name, there must be some way to spread trust anchors and trust policies, e.g., routing protocol, DNS-like infrastructure.

IV. Debates on Naming and Caching

In this section, we mainly describe our understanding on two debatable topics. However, some well-known concerns, e.g., routing scalability, scalability of stateful forwarding, fast name lookup, key management, and producer mobility, are not included in this survey.

4.1 Hierarchical v.s. Self-certifying Names

1) Cache Poisoning of Hierarchical Name: Cache poisoning, referring to the content with given name is faked by attackers and propagated among the network, is a serious problem for hierarchical name. For example, given a specific name, say "www.google.com", if the cached copies are generated by an unauthorized party, and widely distributed among the network, the consumers can merely retrieve the real data without knowing extra information. [7] Thus, it becomes a denial of service (DoS) attack.

As pointed out in preceding section, evidence-based security can solve this problem in the condition that consumers knowing some evidence of target NDO, e.g., its digest, or the digest of the public key that should be used to sign the NDO. And ICN with hierarchical name can seamlessly support this mechanism. Take NDN as an example, there is a Publisher-PublicKeyDigest (PPKD) optional field in the formats of request and NDO, which contains the signature digest of public key just just like the P part in self-certifying names.

2) Provenance and Relevance of Self-certifying Name: Self-certifying name contains two assertions about target NDOs: 1) the public key (or the digest) of producer (P part); and 2) the unique label that electronic content can be mapped to with cryptographic way with give key. These two assertions cannot be provided by the end users (human being) directly. For example, if an end user want to get the main-page of Google, he/she cannot pass a name like "0aldd313:axdeas13" to ICN as parameter of network primitive; even if the name "0aldd313:axdeas13" is in hand, end users cannot decide whether it is the name of desired NDO or not. However, a name like "www.google.com" helps to identify what the data is and choose trust anchor.

The provenance and relevance must be introduced to the trust management in the first place for self-certifying name, e.g., link description on webpages, link sent from personal emails or messages from social networking tools, as suggested by [36]. However, how to ensure that the end users get the"authentic" webpages, emails and messages, or their self-certifying names is remained unsolved; furthermore, this is the requirement of evidence-based security, but cannot fit general

[7] Request can ask for refresh content, and reject cached copy, which helps to ease the problem.

cases. A network infrastructure can be built to address this problem, wherein how to secure the data (validity, provenance and relevance) retrieved from the infrastructure becomes a new challenge.

3) Website Phishing: Phishing happens in the current world-wide web (WWW) where hierarchical name is adopted. [35] argues that ambiguous hierarchical name can be exploited for phishing attack, due to "weak intrinsic name-RWI (real-world identity) binding". For example, name "www.google.com" is mainpage of Google, which is well-known to the public. But what about the following names: "www.google.io", "www.google.com.hk" ? Do those names point to website of Google too? Another example is that, acronym of Bank of America is boa; however, "www.boa.com" may be not owned by the bank. Those names confuse end users, and may be exploited for phishing. Anyway, mapping between the RWI (e.g., Bank of American or its mainpage) and cyberspace identity (e.g., data name "www.boa.com") is purely human's decision and beyond the scope of network protocols, for both hierarchical or self-certifying names, although some names just cannot be mapped to RWIs by human directly, some names make this mapping easy, and some names confuse people.

However, confusing names or not, the data has to be authenticated by a key that can trace back to a trust anchor that the user already knows in the context of hierarchical name. If authentication of a data with name "www.boa.com" ends up with no configured trust anchor, phishing is prevented; or if the authentication traces back to a trust anchor not configured for the bank, which is then noticed by end users[8], phishing is prevented too.

So the essential thing is choosing the proper trust anchor(s) for data names. End users cannot choose trust anchor blindly, otherwise, security cannot be guaranteed in any case. In the context of WWW+HTTPS, root certificates of multiple Certificate Authorities (CAs) are used as trust anchors to certify domain names. Thus, even confusing names can be validated successfully if only the confusing name is certified by one of CAs.[9] In the context of self-certifying name, P can be treated as the trust anchor although it is used to sign the data directly without any hierarchy. P is assumed to be pre-known by consumers in general case. In the context of hierarchical name, applications can choose trust model that is best fitted, e.g., hierarchical trust model, web-of-trust. And end users can configure proper trust anchor(s) for a specific name (or name prefix) without relying CAs to secure the data. What is more, the certificates themselves are named hierarchically, and thus facilitate the trust anchor configuration. Even in this worst case, wherein NDN applications rely on CAs, phishing is not worse than WWW+HTTPS.

4) Naming Debate Summary: Both self-certifying and hierarchical names can identify data independent from the container and session, and security is the main concern.

For hierarchical name, it prevent cache poisoning if some evidences is pre-known, just like the self-certifying name. While for general case, further work is needed for caching poisoning. And website phishing can be prevented by choosing proper trust anchor, wherein certificates with hierarchical names also help.

For self-certifying name, ensuring the prior assertions of provenance and relevance of the name, i.e., the end users must believe that the name they have in hand is the desired data and produced by the right party, is the fatal challenge. Without the prior assertions, the usability and security cannot stand.

4.2 Edge v.s. Pervasive caching

In-network caching of ICN can facilitate the large scale data distribution. Larger cache size means more packets stored and a higher cache hit ratio, but it is meaningless to enlarge the cache size with trivial increasement on hit ratio for business consideration. The cache replacement policies also impact the hit ratio and the performance has been extensively studies, where the simplest one is the LRU. There still exists an open discussion for the

[8] End users may leave some notes for configured trust anchor, which then shows in the browser and reminds end users what the trust anchor should be used for.

[9] In more cases, confusing domain names do not point to servers with HTTPS support.

upper bound of caching hit ratio regardless of caching replacement policies, where the ratio is depend on requests distribution.

This discussion leads to a most debatable topic for the deployment: edge or pervasive caching. Edge caching means placing storage at edge of the network, such as on access routers; while pervasive caching implies that all routers in the network can cache data. Compared to pervasive caching, edge caching simplifies the deployment without upgrading the core routers.

[30] makes use of a request trace dataset from Akamai, the most important CDN provider, wherein requests follow zipf distribution. Based on this dataset, the simulation shows that edge caching is able to achieve nearly the same performance of pervasive caching in terms of query latency, congestion, and maximum origin server load.

On the contrary, [31] concludes a very different conclusion based on the dataset from access and back-haul network of the French ISP, Orange S.A. in Paris. Their observation is that entire request distribution turns out to be trimodal with three components: a discrete Weibull[37] for the head of the distribution, a Zipf for the waist and, a Weibull again for the tail. Based on this dataset, their simulation result shows that pervasive caching gains significant improvement compared to edge caching in terms of traffic reduction.

The opposite conclusion has its theoretical explanation. An overestimation of the catalog size by a given factor under the all-Zipf model would lead to memory over-sizing of the same factor for a given target miss ratio. Conversely, the miss ratio under Weibull requests[37], e.g. of an LRU cache, can be estimated with arbitrary precision by increasing the size of the sample to estimate the popularity law. As we can see, requests distribution is the fundamental factor for caching storage placement. However, this essential issue is not fully clear yet and needs further investment. For the above two studies, dataset in [31] from the real access and back-haul network of large ISP, is more representative than that in [30] from a CDN provider, but neither is real ICN traffic.

V. Summary

In this survey, we illustrate evolution of the Internet, and emphasize the motivations and rationals to develop information-centric networking. We then present the design components of ICN from the perspective of Internet architecture, with briefly summary of the design choices. Furthermore, we analyze two debatable topics in ICN, naming and caching. We achieve the following conclusion based on our analysis:

- We make an clear clarification that, ICN is not about caching only. ICN is a very revolutionary architecture, with the goal of secure, efficient and flexible data delivery. It facilitates large scale data distribution, beyond which ICN could make a difference on a broader scale.
- The advantages of self-certifying name is based on the prior provenance and relevance, which can be supported by hierarchical name as well via evidence-based security under the same assertions.
- The request distribution needs urgent study and analysis, since it is the fundamental factor to study cache replacement strategies and storage placement strategies from the perspective of both theory and practice.
- Generally speaking, ICN is still an ongoing research. There are only a few consensuses on the details of design. We hope this survey helps to settle some disputes, and achieve more consensuses.

Finally, we have to point out the incompatibility between ICN and IP together with millions of IP-based applications, is the cost of introducing those revolutionary elements into layer-3 design and the most crucial obstacle preventing ICN from industrial deployment in current stage. For now, most ICN projects build their network in overlay manner, just like what IP did to telephony in IP's early stage, to facilitate further study.

ACKNOWLEDGMENT

Thanks to Lixia Zhang for many criticisms and discussions that helped clarify the technical descriptions of this paper.

This work is supported by the National High-tech R&D Program ("863" Program) of China (No.2013AA013505), the National Science Foundation of China (No.61472213), and State Scholarship Fund from China Scholarship Council (No. 201406210270).

References

[1] J. Postel *et al.*, "Rfc 791: Internet protocol," 1981.

[2] Cisco, "Cisco visual networking index: Forecast and methodology, 2011–2015," *CISCO White paper*, 2012.

[3] T. Koponen, M. Chawla, B. Chun, A. Ermolinskiy, K. Kim, S. Shenker, and I. Stoica, "A data-oriented (and beyond) network architecture," in *ACM SIGCOMM Computer Communication Review*, vol. 37, pp. 181– 192, ACM, 2007.

[4] N. Fotiou, P. Nikander, D. Trossen, and G. C. Polyzos, "Developing information networking further: From psirp to pursuit," in *Broadband Communications, Networks, and Systems*, pp. 1–13, Springer, 2012.

[5] D. Lagutin, K. Visala, and S. Tarkoma, "Publish/subscribe for internet: Psirp perspective.," *Future Internet Assembly*, vol. 84, 2010.

[6] C. Dannewitz, "Netinf: An information-centric design for the future internet," in *Proc. 3rd GI/ITG KuVS Workshop on The Future Internet*, 2009.

[7] B. Ahlgren, M. D'Ambrosio, C. Dannewitz, A. Eriksson, J. Golic, B. Grönvall, D. Horne, A. Lindgren, O. Mämmelä, M. Marchisio, *et al.*, "Second netinf architecture description," *4WARD EU FP7 Project, Deliverable D-6.2 v2. 0*, 2010.

[8] N. Niebert, S. Baucke, I. El-Khayat, M. Johnsson, B. Ohlman, H. Abramowicz, K. Wuenstel, H. Woesner, J. Quittek, and L. M. Correia, "The way 4ward to the creation of a future internet," in *Personal, Indoor and Mobile Radio Communications, 2008. PIMRC 2008. IEEE 19th International Symposium on*, pp. 1–5, IEEE, 2008.

[9] T. Edwall, "Scalable & adaptive internet solutions (sail)," 2011.

[10] L. Zhang, D. Estrin, J. Burke, V. Jacobson, J. Thornton, D. Smetters, B. Zhang, G. Tsudik, D. Massey, C. Papadopoulos, *et al.*, "Named data networking (ndn) project," tech. rep., Tech. report ndn-0001, PARC, 2010.

[11] V. Jacobson, D. Smetters, J. Thornton, M. Plass, N. Briggs, and R. Braynard, "Networking named content," in *Proceedings of the 5th International Conference on Emerging Networking Experiments and Technologies*, ACM, 2009.

[12] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher, and B. Ohlman, "A survey of information-centric networking," *Communications Magazine, IEEE*, vol. 50, no. 7, pp. 26–36, 2012.

[13] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher, and B. Ohlman, "A Survey of Information-Centric Networking (Draft)," in *Information-Centric Networking* (B. Ahlgren, H. Karl, D. Kutscher, B. Ohlman, S. Oueslati, and I. Solis, eds.), no. 10492 in Dagstuhl Seminar Proceedings, (Dagstuhl, Germany), Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, Germany, 2011.

[14] G. Xylomenos, C. N. Ververidis, V. A. Siris, N. Fotiou, C. Tsilopoulos, X. Vasilakos, K. V. Katsaros, and G. C. Polyzos, "A survey of information-centric networking research," *Communications Surveys & Tutorials, IEEE*, vol. 16, no. 2, pp. 1024–1049, 2014.

[15] G. Tyson, N. Sastry, I. Rimac, R. Cuevas, and A. Mauthe, "A survey of mobility in information-centric networks: challenges and research directions," in *Proceedings of the 1st ACM workshop on Emerging Name-Oriented Mobile Networking Design-Architecture, Algorithms, and Applications*, pp. 1–6, ACM, 2012.

[16] J. Pan, S. Paul, and R. Jain, "A survey of the research on future internet architectures," *Communications Magazine, IEEE*, vol. 49, no. 7, pp. 26– 36, 2011.

[17] B. Fenner, M. Handley, H. Holbrook, and I. Kouvelas, "Rfc 4601: Protocol independent multicast-sparse mode (pim-sm): Protocol specification (revised)," 2006.

[18] A. Ford, C. Raiciu, M. Handley, S. Barre, and J. Iyengar, "Rfc 6182, architectural guidelines for multipath tcp development," 2011.

[19] C. E. Hopps, "Analysis of an equal-cost multipath algorithm," 2000. [20] R. Steward, "Rfc4960: Stream control transmission protocol," 2007. [21] E. Kohler, M. Handley, S. Floyd, and J. Padhye, "Datagram congestion control protocol (dccp)," 2006.

[22] B. Ford and J. R. Iyengar, "Breaking up the transport logjam.," in *HotNets*, pp. 85–90, 2008.

[23] M. Hosseini, D. T. Ahmed, S. Shirmohammadi, and N. D. Georganas, "A survey of application-layer multicast protocols," *Communications Surveys & Tutorials, IEEE*, vol. 9, no. 3, pp. 58–74, 2007.

[24] J. Liang, J. Jiang, H. Duan, K. Li, T. Wan, and J. Wu, "When https meets cdn: A case of authentication in delegated service," in *Security and Privacy (SP), 2014 IEEE Symposium on*, pp. 67–82, IEEE, 2014.

[25] R. Dunaytsev, D. Moltchanov, Y. Koucheryavy, O. Strandberg, and H. Flinck, "A survey of p2p traffic management approaches: best practices and future directions," *Journal of Internet Engi-*

neering, vol. 5, no. 1, pp. 318–330, 2012.

[26] D. Smetters and V. Jacobson, "Securing network content," tech. rep., Citeseer, 2009.

[27] Y. Sun, S. K. Fayaz, Y. Guo, V. Sekar, Y. Jin, M. A. Kaafar, and S. Uhlig, "Trace-driven analysis of icn caching algorithms on video- on-demand workloads," in *Proceedings of the 10th ACM International on Conference on emerging Networking Experiments and Technologies*, pp. 363–376, ACM, 2014.

[28] K. Cho, M. Lee, K. Park, T. T. Kwon, Y. Choi, and S. Pack, "Wave: Popularity-based and collaborative in-network caching for content- oriented networks," in *Computer Communications Workshops (INFOCOM WKSHPS), 2012 IEEE Conference on*, pp. 316–321, IEEE, 2012.

[29] Z. Ming, M. Xu, and D. Wang, "Age-based cooperative caching in information-centric networks," in *Computer Communications Workshops (INFOCOM WKSHPS), 2012 IEEE Conference on*, pp. 268–273, IEEE, 2012.

[30] S. K. Fayazbakhsh, Y. Lin, A. Tootoonchian, A. Ghodsi, T. Koponen, B. M. Maggs, K. Ng, V. Sekar, and S. Shenker, "Less pain, most of the gain: Incrementally deployable icn," 2013.

[31] C. Imbrenda, L. Muscariello, and D. Rossi, "Analyzing cacheable traffic in isp access networks for micro cdn applications via content-centric networking," in *Proceedings of the 1st international conference on Information-centric networking*, pp. 57–66, ACM, 2014.

[32] G. Zhang, Y. Li, and T. Lin, "Caching in information centric networking: a survey," *Computer Networks*, vol. 57, no. 16, pp. 3128–3141, 2013.

[33] Alimi, R. R, Y. A, and Y, "A survey of in-network storage systems," tech. rep., RFC 6392, October, 2011.

[34] G. Carofiglio, G. Morabito, L. Muscariello, I. Solis, and M. Varvello, "From content delivery today to information centric networking," *Computer Networks*, vol. 57, no. 16, pp. 3116–3127, 2013.

[35] A. Ghodsi, T. Koponen, J. Rajahalme, P. Sarolahti, and S. Shenker, "Naming in content-oriented architectures," in *Proceedings of the ACM SIGCOMM workshop on Information-centric networking*, pp. 1–6, ACM, 2011.

[36] D. Trosse, "On long-lived routing identifiers." http://www.fp7- pursuit.eu/PursuitWeb/?p=244.

[37] P. R. Jelenkovic´, "Asymptotic approximation of the move-to-front search cost distribution and least-recently used caching fault probabilities," *Annals of Applied Probability*, pp. 430–464, 1999.

Biographies

JIANG Xiaoke, received the B.S. degree at Tsinghua University, Beijing, China. He is now a Ph.D. candidate of Department of Computer Science, Tsinghua University. His research area is Information-centric Networking (ICN), especially Named-data Networking (NDN). Email: jiangxk10@mails.tsinghua.edu.cn

BI Jun, received B.S., C.S., and Ph.D. degrees in Department of Computer Science and Technology at Tsinghua University. He was a postdoctoral scholar at Bell Laboratories Research and a research scientist at Bell Labs, USA. Currently, he is a full professor and director of Network Architecture & IPv6 Research Division, Institute for Network Sciences and Cyberspace at Tsinghua University. His current research interests include Internet Architecture and Protocols. He is the leading expert (PI) of a major Future Network project supported by China "863" High-tech program: Future Network architecture and INnovation Environment (FINE). He is co-chair of AsiaFI (Asia Future Internet Forum) Steering Group. He served as TPC Co-Chair of ACM SIGCOMM sponsored International Conference on Future Internet, and Co-Chairs of a number of Future Internet related workshops/Tracks at INFOCOM, ICNP, MobiHoc, and ICCCN, especially the Co-Chair of INFOCOM NOM (Named Oriented Mobility) Workshop. He served on Organization Committee or Technical Program Committees of SIGCOMM, ICNP, INFOCOM, CoNEXT, etc. He is a senior member of IEEE, a senior member of ACM, a distinguished member of CCF (China Computer Federation), and a senior member of CIC (China Institute of Communications). The corresponding author, Email: junbi@tsinghua.edu.cn

NAN Guoshun, a Ph.D. candidate in State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications (BUPT), Beijing, China. He received his M.S. in Computer Science and B.S. in Telecommunication Engineering both from BUPT. Then he worked in Hewlett-Packard (HP) as a customer support engineer. His main research interests include the technology of network services, web server optimization, multimedia transmission on future internet. Email: nanguoshun@bupt.edu.cn

LI Zhaogeng, received the B.S. degree at Tsinghua University, Beijing, China. He is now a Ph.D. candidate in Department of Computer Science, Tsinghua University. His research interests include Information-centric networking (ICN), Data-center Network. Email: li-zg07@mails.tsinghua.edu.cn