2-26-2019

Is DNS Ready for Ubiquitous Internet of Things?

Zhiwei Yan
China Internet Network Information Center, China

Hongtao Li
China Internet Network Information Center, China

Sherali Zeadally
University of Kentucky, szeadally@uky.edu

Yu Zheng
China Internet Network Information Center, China

Guanggang Geng
China Internet Network Information Center, China

Is DNS Ready for Ubiquitous Internet of Things?

Abstract

The vision of the Internet of Things (IoT) covers not only the well-regulated processes of specific applications in different areas but also includes ubiquitous connectivity of more generic objects (or things and devices) in the physical world and the related information in the virtual world. For example, a typical IoT application, such as a smart city, includes smarter urban transport networks, upgraded water supply, and waste-disposal facilities, along with more efficient ways to light and heat buildings. For smart city applications and others, we require unique naming of every object and a secure, scalable, and efficient name resolution which can provide access to any object's inherent attributes with its name. Based on different motivations, many naming principles and name resolution schemes have been proposed. Some of them are based on the well-known domain name system (DNS), which is the most important infrastructure in the current Internet, while others are based on novel designing principles to evolve the Internet. Although the DNS is evolving in its functionality and performance, it was not originally designed for the IoT applications. Then, a fundamental question that arises is: can current DNS adequately provide the name service support for IoT in the future? To address this question, we analyze the strengths and challenges of DNS when it is used to support ubiquitous IoT. First, we analyze the requirements of the IoT name service by using five characteristics, namely security, mobility, infrastructure independence, localization, and efficiency, which we collectively refer to as SMILE. Then, we discuss the pros and cons of the DNS in satisfying SMILE in the context of the future evolution of the IoT environment.

Keywords

DNS, TCP/IP, Internet, IoT, name service

Disciplines

Computer and Systems Architecture | Digital Communications and Networking

Notes/Citation Information

Published in *IEEE Access*, v. 7, p. 28835-28846.

IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 25, 2019, accepted February 10, 2019, date of publication February 26, 2019, date of current version March 18, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2901801

Is DNS Ready for Ubiquitous Internet of Things?

ZHIWEI YAN[1], HONGTAO LI[1], SHERALI ZEADALLY[2], YU ZENG[1], AND GUANGGANG GENG[1]

[1]China Internet Network Information Center, Beijing 100190, China
[2]College of Communication and Information, University of Kentucky, Lexington, KY 40506, USA

Corresponding author: Guanggang Geng (gengguanggang@cnnic.cn)

This work was supported by the National Standardization Project for Intelligent Manufacturing—research and verification of IPv6-based standard for industrial Internet.

ABSTRACT The vision of the Internet of Things (IoT) covers not only the well-regulated processes of specific applications in different areas but also includes ubiquitous connectivity of more generic objects (or things and devices) in the physical world and the related information in the virtual world. For example, a typical IoT application, such as a smart city, includes smarter urban transport networks, upgraded water supply, and waste-disposal facilities, along with more efficient ways to light and heat buildings. For smart city applications and others, we require unique naming of every object and a secure, scalable, and efficient name resolution which can provide access to any object's inherent attributes with its name. Based on different motivations, many naming principles and name resolution schemes have been proposed. Some of them are based on the well-known domain name system (DNS), which is the most important infrastructure in the current Internet, while others are based on novel designing principles to evolve the Internet. Although the DNS is evolving in its functionality and performance, it was not originally designed for the IoT applications. Then, a fundamental question that arises is: can current DNS adequately provide the name service support for IoT in the future? To address this question, we analyze the strengths and challenges of DNS when it is used to support ubiquitous IoT. First, we analyze the requirements of the IoT name service by using five characteristics, namely security, mobility, infrastructure independence, localization, and efficiency, which we collectively refer to as SMILE. Then, we discuss the pros and cons of the DNS in satisfying SMILE in the context of the future evolution of the IoT environment.

INDEX TERMS DNS, TCP/IP, Internet, IoT, name service.

I. INTRODUCTION

Advances in communication technologies have paved the way for ambient environments where most electronic devices are able to connect to ubiquitous networks [1]. Integrated networks connecting a large number of objects with different functions, form the so-called Internet of Things (IoT) [2]. The basic motivation behind IoT is to enable these objects (or things, devices) to communicate with each other and exchange information in various contexts [3].

IoT will enable traditional informational applications to be more intelligent and efficient. These independent applications are increasingly being integrated into a smart and comprehensive system to improve the daily lives of people and digitize city management thereby paving the way for the emergence of smart city as a typical application area of IoT [4]–[6]. In order to develop such a complex IoT based informational system, the design of scalable, robust, secure and efficient infrastructure including especially name and address services is vital. In addition, IoT is being extended to a wider concept, often defined as the Internet of Everything (IoE) [7], [8] which extends business and industrial processes further and it is composed of people, data, processes and things [9], [10]. A recent report by Gartner forecasted that 20.8 billion IoT devices will be deployed by 2020 [11]. IoT will bring billions or more objects to the public network. Therefore, as we evolve from the web-based Internet to object-based IoT and ultimately IoE, we will need to address new challenges to the name and address services that are currently in use.

For the address service, the Internet Protocol version 6 (IPv6) makes it possible to address the huge number of objects which potentially need to be accessed. IPv6 follows the Internet Protocol Version 4 (IPv4), which can only support 2^{32} IP addresses and they were almost exhausted from the perspective of the Internet Assigned Numbers Authority (IANA) in about 2011 (the last two unreserved IANA/8 address blocks have been allocated) [12]. In contrast,

The associate editor coordinating the review of this manuscript and approving it for publication was Miltiadis Lytras.

IPv6 enables up to 2^{128} IP addresses which are adequate for addressing the increasing number of objects joining the Internet. Besides, IPv6's improvements on end-to-end connectivity, security and mobility are also attractive features, along with its improved scalability of course. In order to make IPv6 more suitable for resource-restricted IoT devices, IPv6 over Low-Power Wireless Personal Area Networks (6LoWPAN) compresses the IPv6 address into a couple of bytes [13]. And with the massive IPv6 address space, two networked devices can communicate with each other directly with public IPv6 addresses, without the necessary of Virtual Private Network (VPN) to reserve the IP address resource. Various communities around the world continue to push the Internet to evolve from IPv4 to IPv6 in order for the addressing service to be ready for the future IoT era and the deployment of IPv6 has increased dramatically during recent years.

However, for the name service, multiple solutions have been proposed and the most popular example is the Object Name Service (ONS) for the Electronic Product Code (EPC) identification [14] and it was proposed by the Auto-ID center. It can provide a mapping service between the RFID tag's EPC and the address of the EPC information server. Besides, ONS was also extended to support the discovery service in order to integrate more information associated with an EPC. Currently, ONS is widely used in logistics-related applications globally and it is the most successful name service in IoT. In addition, there are many other name services for different applications such as the Handle system for the digital object management [15], Named Data Networking (NDN) for the named content [16], MobilityFirst for the Global Unique Identification (GUID) [17], NovaGenesis [18], [19] to support any combination of flat or hierarchical, natural language or self-verified naming scheme and so on. Within these proposals, some are designed based on the Domain Name System (DNS) [20], [21] which is widely used for service naming in today's Internet, while some are based on clean-slate models to facilitate the new requirements in the future Internet. As the fundamental infrastructure in the current Internet, DNS was standardized early on to provide the hierarchical name management and resolution service in the global Internet or a private network. With distributed databases maintaining tree-structured name zones, DNS is usually used to map human-readable host names to IP addresses which are used by network devices (e.g., routers) and vice versa.

Moreover, DNS is evolving continuously in terms of both its service scale (e.g., number of names and number of servers) and protocol functions. Since the emergence of DNS over 30 years ago, the total number of domain names only at the second level has exceeded 300 million currently. Additionally, millions of servers (including both authoritative servers and recursive servers) have been deployed globally to support the stable resolution of this huge name space. Then, one fundamental question arises: *will the current Internet DNS be able to support the object naming and name resolution in the ubiquitous IoT which may have more sophisticated requirements for the name service?* To answer this question, the remainder of this paper is organized as follows. First, we summarize major recent DNS developments. Next, we analyze the most common requirements of the IoT name service. We then study the pros and cons of DNS to support the IoT name service and, finally, we present some concluding remarks.

We summarize the contributions of this work as follows:

- We analyze some typical IoT applications and the key requirements of the name service. Accordingly, we propose the SMILE which embodies the five most important and common requirements for an IoT name service.
- We analyze the functionality and performance of DNS and its current deployment model to demonstrate its pros and cons in satisfying the SMILE requirements.
- We summarize the various name services both based on DNS extensions and clean-slate designs which could be used in different application scenarios, to illustrate the evolution of the future IoT name service.

II. EVOLUTION AND DEVELOPMENT OF DNS

The identification of a domain name in the Internet includes two properties: brand label and protocol identifier. A name that can be more easily-remembered (e.g., *z.cn*, *vip.com*) can better illustrate the brand ownership and thus it has higher value in the DNS market when it shows the brand label property of a name. On the other hand, the name works as the interface through which the Internet user accesses the related application or service. From a unique domain name, the DNS resolution service provides the information of the specified application or service, which shows the protocol identifier property of the domain name. Corresponding to the two properties of a domain name mentioned above, there are two business procedures in DNS (as illustrated in Fig. 1). If one registrant wants to register a domain name under a selected Top-Level Domain (TLD), he/she needs to apply for this name through a registrar who has been authorized by the related registry to sell out the names under that TLD. During this procedure, the Extensible Provisioning Protocol (EPP) [22] is used. To resolve a domain name, the DNS resolution procedure is used. In the recursive resolution model, a client (requester) sends the request message out and the request is processed directly by the default recursive server. Based on recursive logic and local cache, the response containing the related answer will finally reach the requester.

On the right side of Fig. 1, the TLD management procedure is managed by the Internet Corporation for Assigned Names and Numbers (ICANN) and the Public Technical Identifiers (PTI) which currently performs the IANA functions on behalf of ICANN. Next, some contracts and specifications are followed by these entities (e.g., registry, ICANN, PTI), and the name related data is also synchronized between different databases to provide the further resolution and Whois service [23]. With the increasing requirements from the community to extend the TLD space to enhance competition, innovation, and consumer choice, in 2012,

TABLE 1. MAIN DNS RRS.

RR Type	Meaning	RR Type	Meaning
A	IPv4 address	RP	Responsible person
AAAA	IPv6 address	SIG	Signature
AFSDB	AFS database	SOA	Start of authority
APL	Address prefix list	OPT	Option
CERT	Certificate	SRV	Service locator
CNAME	Canonical name	SSHFP	SSH public key fingerprint
DHCID	DHCP identifier	TKEY	Transaction key
DLV	DNSSEC lookaside validation	TLSA	TLSA certificate association
DNAME	Alias for a name and all its sub-names	TSIG	Transaction signature
DNSKEY	DNS key	TXT	Text
DS	Delegation signer	NAPTR	Naming authority pointer
HIP	Host identity protocol	NS	Name server
IPSECKEY	IPsec key	NSEC	Next secure
KEY	Key	NSEC3	Next secure v3
KX	Key exchanger	*	All cached records
LOC	Location	PTR	Pointer
MX	Mail exchange	RRSIG	DNSSEC signature
AXFR	Authoritative zone transfer	IXFR	Incremental zone transfer

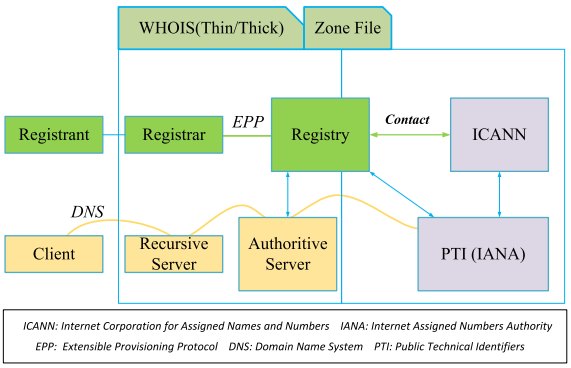


FIGURE 1. DNS business logic.

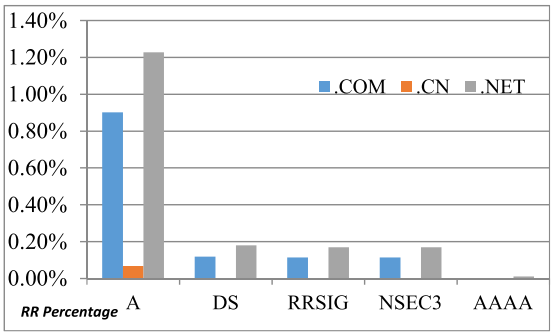


FIGURE 2. Mostly used RRs (without NS RR) in.CN/.COM/.NET zones.

ICANN launched the new gTLD application plan and until now more than 1200 new gTLDs with many Internationalized Domain Name (IDN) strings have been added into the DNS root zone [24]. To support the DNS evolution with the emerging requirements of secure and scalable name services, the Internet Engineering Task Force (IETF) keeps extending the basic DNS protocols. Dozens of Resource Records (RRs) have been proposed to provide different kinds of domain name related information and Table 1 lists some main RRs which are actively supported by the current DNS, although in practice, these RRs are not fully used when we analyzed the zone files of.CN,.COM and. NET. The RRs that are mainly used under these zones are shown in Fig. 2.

For the zones of.CN,.COM and. NET, 98% of RRs used are NS which is used to delegate the name server of the related subdomain. Except for NS RR, the RRs ranked in the top 5 are A RR for the IPv4 address of the entity, AAAA for the IPv6 address of the entity, and DS, RRSIG and NSEC3 are the Domain Name System Security Extensions (DNSSEC) [25]–[27] related RRs. DNSSEC is implemented based on the basic DNS protocol to help guaranteeing DNS data origin authority, integrity, and authenticated denial of existence. Based on these aforementioned results, we note that today, DNS can adequately support many application scenarios and protocol extensions to locate different kinds of resources and information. In addition, based on DNSSEC, the DNS RRs can be secured and some traditional security architecture based on the Public Key Infrastructure (PKI) can be implemented on DNS/DNSSEC to issue the certificate associated with a domain or to

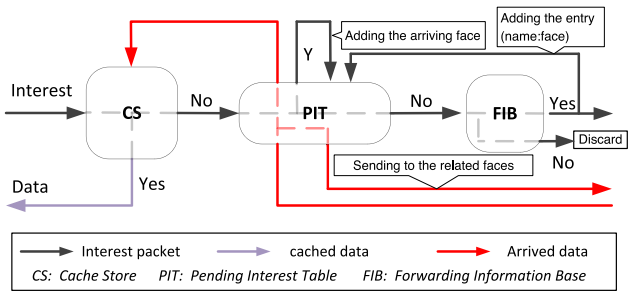


FIGURE 3. NDN communication principle.

maintain the key related information binding with the specified name [28].

Then, on one hand, it is better to make good use of the current DNS function, and on the other hand it is preferable to avoid introducing more RRs into the DNS especially those that are complex but with limited functions.

III. SMILE: IoT NAME SERVICE REQUIREMENTS

A. SECURITY

In the ubiquitous IoT, more devices will be connected which means that there will be more vulnerabilities that can be exploited by hackers. As predicted, 70% of IoT devices are not securely deployed and are vulnerable to different attacks by 2020 [29]. From this point, security must be the foundational enabler for IoT.

Moreover, as the most important infrastructure in IoT, the name service must be robust and secure enough to support stable and trusted name management for the huge number of smart objects. Generally, the name service has to guarantee the robustness of the service (e.g., defend for the attacks such as Distributed Denial of Service (DDoS)), support the verification of the data to prevent the spoofing and tampering of the transmitted registration or resolution data, and to protect the privacy of the name-related information because IoT names may carry some sensitive information about the end-user. For the first requirement (to guarantee the robustness of the service), the system deployed should be scalable enough and not binding to other system to maintain its service ability. While for the latter two requirements (to support the verification of the data and to protect the privacy of the name-related information), some mature credential technologies should be integrated in the naming scheme or/and the name resolution service.

To enhance security, Named Data Networking (NDN) re-designed the TCP/IP based communication model as shown in Figure 3 [16], [30]. A data object in NDN is segmented into multiple small pieces which can be individually named and transmitted in independent transactions. The content name in NDN is hierarchically structured and it has a format similar to the DNS domain name.

When a receiver/requester needs some content, it sends out an Interest packet which only specifies the name of the desired content. The router which receives the Interest packet checks its Cache Store (CS) with the content name as an index. If there is a positive match, the router directly sends back the related content in a Data packet to the interface which received the Interest packet. Otherwise, the router chooses an outgoing interface to send out the Interest packet based on its Forwarding Information Base (FIB) which follows the longest prefix match principle similar to the way the IP routing table works. The router propagates the unmatched Interest packet and it records the content name and incoming interface of the Interest packet in its Pending Interest Table (PIT). In this way, the router can make sure that the future matched Data packet can be routed back to the receiver along the reverse path.

As the data objects are independently routed in a distributed manner in NDN, a key is usually bound to a content to encrypt the content in order to guarantee the content's integrity or privacy. Compared with the basic security model in the current TCP/IP network, the efficiency of the NDN security scheme could be improved as it avoids the pre-establishment of a security channel.

However, the management of NDN names is still not well defined at present [31]. For actual deployment, especially when security must be considered for the IoT name service, the distributed delegation and the flexible trust model based on the name are essential parts to be considered. The former one (i.e., distributed delegation) can be used to establish the automatic content management architecture to support the distributed name-based routing. The latter one (i.e., flexible trust model) is used to manage name based keys in order to satisfy the different security requirements (e.g., data integrity, data origin authentication and privacy protection) in NDN.

B. MOBILITY

The integration of the traditional telecommunication system and the Internet has paved the way for a plethora of mobile Internet applications. As Cisco Visual Networking Index (VNI) predicted, the number of global mobile users will be over 5.5 billion by 2020, and this will represent 70% of the population of the world. Additionally, 98% of mobile data will be generated by different smart devices [32].

Mobility has always been challenge for IP even before the emergence of IoT. Mobility protocols such as Mobile IPv6 (MIPv6) [33] and Proxy Mobile IPv6 (PMIPv6) [34] operate at the IP layer and they can handle mobility with the address mapping and IP tunneling-based redirection procedures, to maintain reachability even when the device moves and configures a new locator (e.g., IPv6 address). However, the cost is too high for IoT considering the limited resource of the device, the large number of devices and induced sub-optimized communication path. One possible approach is to update the name mappings directly when the locator changes. It is also cumbersome because the information update in mobile scenario must be efficient enough for the new communications and moreover, another rendezvous point may still be needed to redirect any ongoing communications.

For IoT, mobility is also a basic requirement because many devices will be portable and mobile. We can foresee that evolving from the web-based Internet to object-based IoT will bring new challenges to mobility management solutions. For example, in the smart health care system, many portable sensors and communication modules are carried by people as they move around (when they are walking, driving or taking the public transport system) to monitor their blood oxygen saturation, body temperature, blood pressure and then communicate with the remote doctor or hospital server so that the necessary and timely diagnosis can be provided when needed. In another scenario, both the data requester and the publisher may move during the ongoing communication between them. On one hand, the device may configure a new locator when it moves, on the other hand, the device may configure a new name if the domain being accessed has specified a new name prefix. In addition, both the name and the locator of the device may change and then the mobility management will be more complex. Motivated by the mobility requirement, MobilityFirst [17] separates the name from the network locator to boost mobility performance. MobilityFirst adopts Global Unique IDentification (GUID) as its naming scheme which is designed mainly based on the principle of locator and identifier split. It is not a novel concept. Locator/Identifier Separation Protocol (LISP) [35], [36] also applied this concept but it mainly aims to support mobility and enhance security in the existing TCP/IP architecture. But based on the split between the locator and the identifier, the MobilityFirst introduces both GUID-based routing and locator-based routing for different application scenarios [37].

NDN also provides a solution for the mobile receiver because only the fixed name is used for the communication. However, publisher mobility support is difficult for NDN. Compared with NDN, MobilityFirst reduces the mobility overhead but the routing rules and management system are more complex. Overall, the name service should be flexible and efficient enough to support the dynamic update of the name or the name related information. Additionally, some lightweight and distributed mobility solution should work together with the name service.

C. INFRASTRUCTURE INDEPENDENCE

In IoT, the network topology is more dynamic and it may introduce many resource-constrained and function-specific smart devices. They may include data generation devices, data transmission devices, data analyzing devices and so on. These devices provide different types of services to various smart IoT application scenarios, such as smart home, smart transportation, smart healthy, smart grid and others [38], [39].

To support the communication of inter-connected smart devices, zero-configuration networking (zeroconf) [40] was developed by IETF. Basically, zeroconf supports the device to configure network address, resolute names and discover services on its own [41]. In this case, name is a basic index for the information retrieval and communication peer detection in the ad-hoc IoT network. However, as the number of IoT devices increases, the manual assignment of names becomes a cumbersome task. Some scheme must be developed to support a task such as name auto-configuration once the device is online and the name should follow a consistent construction principle to avoid collision and this name should also be human-readable as a friendly interface between the user and the device.

In oneM2M [42], the Object Identifier (OID) is adopted. That is, the M2M OID consists of the management-level information and device-level information. Since this auto-configured name contains a lot of detailed management information of a device, users can easily identify a specified device.

Except for the name auto-configuration scheme, the name resolution also should not depend on the infrastructure because sometimes the infrastructure is not available to provide the name resolution. For example, in the infrastructure-less network, how to discover a specified service (e.g., locate the printing server in the local network) or device (e.g., locate a device which can provide the required information) is a very common need in IoT.

D. LOCALIZATION

In IoT, there will be a large number of devices in different application scenarios. Although not all of them need to be globally reachable or even configured with an IPv6 address, the network architecture must support their scalable and efficient inter-connection and global reachability. No matter what the case may be, scalable device management based on the special requirements remains a challenge. And then localization in IoT includes both the working scope and local usage habit requirements.

For the requirement of working scope localization, IoT devices may only work in the restricted domain and cannot or should not be seen externally. For example, in the basic intelligent agriculture system, different sensors will be deployed: to monitor the temperature, humidity, air condition and other environmental parameters. Based on the collected measurement data, the controller will then adjust the environment to guarantee the optimized crop health. In this case, the sensors are only accessible by the controller which is directly connected to them. In this application environment, only a local network should be established without global IPv6 addresses. But the name must be used to organize the sensors and the related data even with local logic.

For the usage habit localization requirement, the device in the IoT should match the user habit in different geographical regions. For example, if someone resides in China and he/she has set his/her smart home to perform certain actions when he/she is at a set distance from his/her home; when approaching closely to his/her house, the heating can be activated for example. But if the heating equipment is made in Japan, it may not understand Chinese instructions. Other factors that affect the user's habits include: time zone, measurement units, currency and so on, and localization helps to adapt these factors to the related location of the device.

In summary, localization in IoT restricts the access of an object to a particular area. And it makes a product adaptable to a specific location or market, to support the product and its related service to be deployed and used without worrying about linguistic, cultural or religious differences. For the name service, the names must also support localization which implies two requirements: 1) the names are only available and meaningful in the specified domain/area; 2) the name labels used in IoT applications can facilitate the local usage habit (e.g., with local language) to enable usage by the local users.

E. EFFICIENCY

Efficiency is very important for latency-sensitive IoT services. For example, the smart grid must execute real-time data collection and control. In the case of smart transportation, the vehicle needs to gather the traffic status rapidly and continually in order to dynamically schedule the optimum route to its destination [43]. Other than the traditional IoT areas which have high efficiency requirement, recent IoT developments and human-computer interaction technologies, such as tactile networks, have revolutionized the entertainment market by the information exchange methods and network architectures. But they also require higher communication efficiency because more data should be handled, exchanged and processed within strict latency constraints.

For the address system, the efficiency of IPv6 based communication can be improved compared with that of IPv4 because end-to-end communication is possible and some in-path devices, such as Network Address Translation (NAT) [44] is unnecessary. To adapt IPv6 for IoT applications, IETF also standardized the 6LoPAN/6lo [45] protocol suite which will facilitate IPv6 connectivity over resource-constrained nodes [46].

For the name service, a more efficient architecture needs to be redesigned. PURSUIT (Publish-Subscribe Internet Technology) [47] is a clean-slate architecture to support efficient content publish and subscribe in the content-centric concept. The two core management elements in the PURSUIT architecture are the rendezvous system and the topology manager. When a publisher holds some content and the publisher wants to share the content, the publisher will publish the related information to the rendezvous system. Once a subscriber/requester has interest to fetch this content, it sends a request to the rendezvous system. Next, the topology manager is called to calculate the optimum transmission path as a Bloom filter string between the subscriber and the publisher [48]. In this way, the forwarding is very simple and efficient in PURSUIT, because the router involved only needs to check which one of its next-hop neighbor is on this forwarding path, and then it deterministically sends the content to the next hop. To adapt to this routing scheme, the content name used in PURSUIT has a format with two parts namely, the rendezvous identifier and the scope identifier.

In short, the name configuration and name resolution in IoT applications must be efficient enough to support more real-time and delay-sensitive applications so that the overall IoT performance will not be affected.

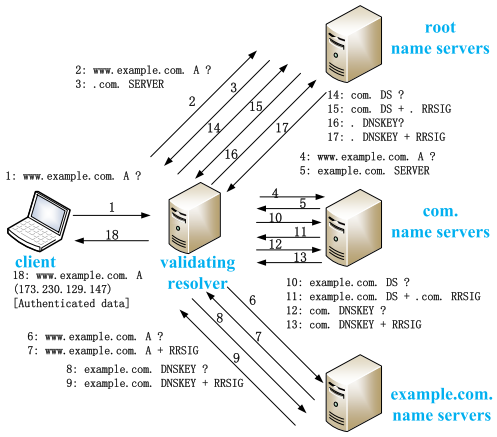


FIGURE 4. DNSSEC procedure.

IV. DNS ADAPTIONS AND CHALLENGES TO SMILE

The emerging requirements of IoT call for an enhanced name service which is fundamental to the overall IoT architecture. As different application models between the traditional Internet and the ubiquitous IoT have emerged over the last few years, we need to analyze the suitability of DNS in meeting the IoT requirements discussed above.

A. SECURITY

DNS was originally designed as an ''open'' protocol. Therefore, it is vulnerable to attackers in almost all its service entities and message exchange procedures. Typical risks include foot-printing, Denial-of-Service (DoS), redirection and so on [49].

To mitigate these risks and enhance the security of the traditional DNS service, the IETF standardized the DNSSEC protocol [25]–[27], [50]–[52]. When an asymmetric-key cryptographic algorithm is used, a DNSSEC-enabled server generates a pair of keys wherein the private key is used to sign the RRs while the public key is used to verify the signed data to make sure the RRs came from the authorized source and was not tampered during transmission. Fig. 4 shows the detailed procedure of DNSSEC.

In order to make this work, some new RRs are introduced: RRSIG contains the signature of the plain RR, DNSKEY is a public key used to generate the RRSIG and DS is used to verify the DNSKEY from the perspective of a parent zone. For example, when the validating resolver receives the RRSIG data with the related A RR of *www.example.com* in step 7, the validating resolver can verify the signature with the related DNSKEY from step 9. However, if the validating resolver does not trust this DNSKEY, the validating resolver can verify this DNSKEY with the corresponding DS record from the upper-level domain as step 11 shows. In this way, a trust chain is constructed from the lower-level zone to the

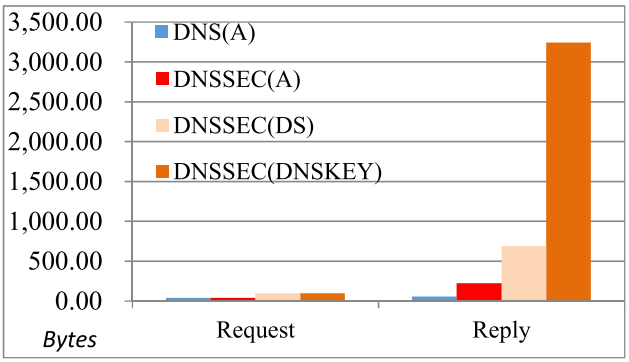


FIGURE 5. Size of signaling messages of DNS with and without DNSSEC.

higher-level zone and ends at the root zone as the trust anchor of DNSSEC.

DNSSEC enhances the security of the plain DNS but the procedure is more complex. To illustrate the cost involved during DNSSEC procedure, we tested a domain name (*www.isc.org*) with DNSSEC-enabled recursive server (i.e., the validating resolver) and the sizes of all the signaling messages transferred are shown in Fig. 5. The basic signaling messages in DNS are only a simple request and a simple reply pair for A RR resolution. However, DNSSEC introduces not only the signature of A RR reply but also some new RRs and procedures (such as DNSSEC(DS) and DNSSEC(DNSKEY) shown in Fig. 5). In this way, compared with the plain DNS protocol, DNSSEC may increase the size of DNS packet to more than 10 times and includes additional procedures to fetch the key information (e.g., DS and DNSKEY).

Based on the trust model established by DNSSEC, the traditional PKI can be implemented in DNS and was subsequently standardized as the DNS-based Authentication of Named Entities (DANE). It is proposed in RFC 6698 [53] to authenticate Transport Layer Security (TLS) client and server without the traditional Certificate Authority (CA). This application scenario is described in detail in RFC 7671 [54]. Moreover, application scenarios of DANE are also described in RFC 7672 [55] for Simple Mail Transfer Protocol (SMTP) and RFC 7673 [56] for Service (SRV) record.

Although DNSSEC can prevent tampering of the DNS data while in transit, the basic DNS protocol still transmits data without any privacy protection. As a result, information about the DNS user is still accessible. IETF then established the DNS PRIVate Exchange (DPRIVE) Working Group [57] to develop mechanisms that provide confidentiality between DNS users and resolvers, and it may also later consider mechanisms that provide confidentiality between resolvers and authoritative servers, or guarantee end-to-end confidentiality of DNS transactions.

Although DNS security has been improved over the years, the costs of both PKI-based DNSSEC and TLS-based DPRIVE are very high. As a result, the DNS reply message with DNSSEC enhancement may be too large to fit into one User Datagram Protocol (UDP) packet. In this case, the DNS server has to send a truncated reply message and the user will

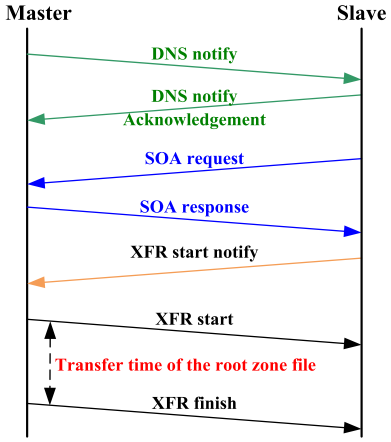


FIGURE 6. Dynamic update between a master node and a slave node.

re-send the reply message by using the Transmission Control Protocol (TCP) or other protocol extensions to support the transmission of larger reply messages [58]. This will be a big challenge for the resource-restricted IoT scenarios. And the current DNS security suite needs too much additional support (e.g., trust model, key management architecture, TLS maintenance) for the IoT applications. In short, a lightweight security model but not the DNSSEC style scheme is needed for an IoT name service.

B. MOBILITY

Even in DNS, the data is dynamic, and the dynamic update protocol [59], [60] supports both single RR and a whole zone file update. When a mobile IoT device changes its IPv6 address or changes its name or changes both its IPv6 address and its name, an update message must be sent out directly to the authoritative server which is in charge of the zone containing the changed information. Next, the master server synchronizes the updated information with all the slave servers. Fig. 6 shows the typical DNS update procedure between the master server node and the slave server node [61]. According to the DNS update protocol, the update of the data does not need to be immediately executed, especially when multiple slave servers need to be updated. This is not acceptable for a mobility solution because changes of the name information must propagate as fast as possible.

Another limiting factor for mobility support by using DNS is the impact of caching in the recursive servers. Normally, DNS data is cached by the recursive server in order to handle future requests efficiently, and the lifetime of the cached data is determined by the Time-To-Live (TTL) field in the AAAA RR for example. To prevent unreachability which may be caused by stale data in the recursive server, the RR of a mobile node whose name or/and address will be dynamic should be set to zero. In this way, the record must be re-fetched from the authoritative server even if it has just been requested by the same recursive server and this will degrade the efficiency of the DNS resolution process.

As illustrated above, using DNS to directly support device mobility is not efficient and scalable [62], although DNS can be used with some mobility protocols (e.g., MIPv6 and PMIPv6) operating at the IP layer to support the communication connectivity of the IoT devices with dynamic name or/and address. But in these solutions, DNS is not used as the database to manage the IoT name-address mappings directly. Instead it is used as a service discovery scheme to locate the database which maintains the device information (e.g., the Home Agent (HA) in MIPv6 or the Local Mobility Anchor (LMA) in PMIPv6). However, if the name in some IoT applications changes, a new model to support efficient mobility management should be proposed [63].

C. INFRASTRUCTURE INDEPENDENCE

Basically, the user needs to request the DNS data with its configured recursive server which acts as a relaying entity or proxy server for the user. The DNS along with the Neighbor Discovery Protocol (NDP) [64] and the Dynamic Host Configuration Protocol (DHCP) [65], can support the initial configuration of the device being accessed with basic parameters. To extend the scope of the name service in the local domain without the available DNS infrastructure, DNS-based Service Discovery (DNS-SD) [66] over Multicast DNS (mDNS) [67] was standardized and it is widely used nowadays for discovery and resolution of services on a zero-configuration local link [68], [69].

DNS-SD is mainly used to discover the names of specified services with standard DNS procedure. The service entity is configured with DNS SRV RR and TXT RR, and the SRV RR specifies the domain name of the service entity and the TXT RR contain more parameters describing the service entity. When a device requests a service entity with PTR RR and the specified service type, the server replies with the SRV RR and TXT RR which contains enough information for the device to request further information of a selected service entity.

When there is no central server on the local link, mDNS can be used together with DNS-SD to discover the service entity. Every device on the multicast link listens on the port 5353 which is the mDNS protocol port number. The request message is sent to a well-known multicast address. The request message can be listened by all the local link devices and the matched one will actively replies to the requester. It means that mDNS allows a network device to choose a domain name in the local DNS namespace and announces it using a special multicast IP address if the name configuration can be automatic. In addition, all the mDNS resolutions can be executed without the help of any infrastructure [70]. However, when a device moves to a multi-link network, DNS-SD/mDNS does not work across routers on different links. In this case, some recent efforts [71] have investigated to extend the DNS-SD to the multi-link scenarios to support infrastructure-independent name service in a more generic manner.

D. LOCALIZATION

To optimize the performance of DNS resolution, localization is enhanced for both recursive and authoritative services. For example, a recursive service can be developed and deployed based on the local requirements. In this way, the location and server configuration can all be determined by the related operator. Basically, the recursive cache consists of two components: the online cache is used to send feedback to users directly and the offline cache is used as a backup to recover the DNS service in case of an emergency [72].

This model is workable because the DNS data integrity can be guaranteed by DNSSEC. For example, at the root level, although the distribution of root servers through anycast is mainly determined by the 12 Root Name Server Operators (RNSOs), the locally controlled recursive service is attempting to play an important role in the localization of the root service. These efforts include the RFC 7706 [73] which makes use of the loopback address in the recursive server to resolve the proactively-fetched root zone, and RFC 8198 [74] which makes use of DNSSEC NSEC/NSEC3 RRs to support a recursive server to generate negative answers within a range and positive answers from wildcards. Then the approach proposed in RFC 8198 improves efficiency on both authoritative and recursive servers, and increases the DNS system's ability to resist DDoS attacks.

Sometimes, the geolocation information should be integrated with DNS to optimize the location-based service in the local area and this can be implemented in two ways, The first way is in the DNS name configuration: when an object configures a name, it can combine the geolocation data in the name and this can support the data aggregation or service discovery from the name service level. The other way is that the DNS protocol supports the maintenance of geolocation information related with a name such as a LOC RR.

From another perspective on localization (local usage habit adaptation), the ICANN started considering IDN [75] in about 1996. After much discussion, Internationalizing Domain Names in Applications (IDNA) [76] was adopted finally and native language scripts have been introduced in more TLDs. In May 2010, the first IDN ccTLD was installed in the DNS root zone. This extension is a very important step for the ubiquitous name service in IoT because local language (e.g., Chinese, Russian, and Japanese) can be used in the DNS name labels and facilitate the management of IoT objects by the user in different language regions. But IDN names need the extension of the application software to convert between ASCII and non-ASCII forms of a domain name. Moreover, widely-used applications and related software (e.g., mail service, browser service) must support this mapping function in a more generic way in order to support the IDN names to be used in localized IoT applications.

E. EFFICIENCY

To guarantee the scalability of the name space, DNS manages the domain names in a hierarchical manner in which the

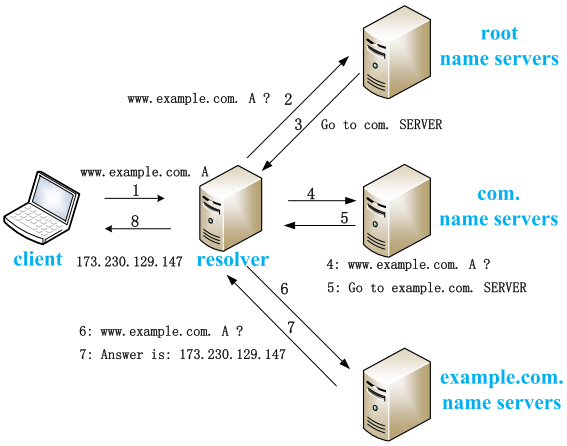


FIGURE 7. DNS resolution process.

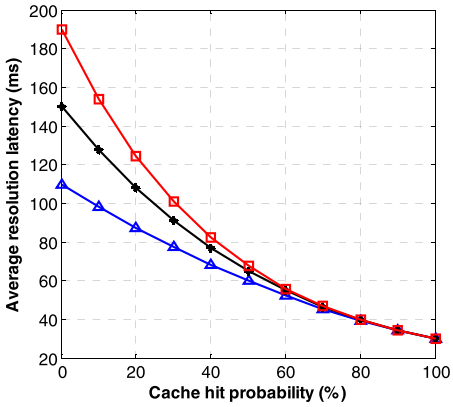


FIGURE 8. DNS resolution latency.

upper-layer delegates names to the lower layer which also guarantees the fault-tolerance of DNS data management in a distributed manner. The resolution of a domain name also follows the delegation relationship between the upper-layer name and the lower-layer name as shown in Fig. 7. The client sends out a request message to its default local recursive server (resolver) which checks its cache to see whether this request can be locally answered. If this check is negative, the resolver needs to request this name to the authoritative servers level-by-level until it receives a comprehensive answer.

Then we can model the DNS resolution latency as follows (we assume that the resolver maintains the hint file when it boots up and the resolution name is at least under TLD as a meaningful second level name):

$$\begin{aligned} L_{DNS} &= p \times R_c + p \times (1-p) \times (R_c + R_r) \\ &\quad + p \times \sum_{i=2}^{n-1} (1-p)^i \times (i \times R_r + R_c) \\ &\quad + (1-p)^n \times (n \times R_r + R_c) \end{aligned}$$

in which p denotes the probability of a hit from the resolver's cache of a name. The probability at each level is independent and identically distributed. R_c denotes the Round-Trip Time (RTT) between client and the resolver, and R_r denotes the RTT between the resolver and the authoritative server. And n is the depth of the domain name.

Based on this model, Fig. 8 shows the average latency for a DNS resolution.

Fig. 8 shows that the average resolution latency for the names at different levels where we set R_c and R_r to 30 ms and 40 ms respectively. As shown, the latency increases when the DNS resolver cannot find the name in its local cache and when the name level increases. When we consider the dynamic characteristic of the large number of names in IoT, latency becomes a major challenge for the applications with strict delay requirements. This only shows the resolution latency of the basic DNS protocol but if under DNSSEC (as shown in Fig. 4), more procedures and larger signaling packets are introduced and the efficiency of the DNS resolution decreases further.

Most DNS transactions use UDP for efficiency whereas TCP is always used for full zone transfers (using AXFR) and is often used for messages whose sizes exceed the DNS protocol's size limit (when truncation occurs). However, the growing deployment of DNSSEC and IPv6 has increased response sizes and therefore the use of TCP. Besides, TCP use has also been increasingly driven by the protection it provides against address spoofing and therefore exploitation of DNS in reflection/amplification attacks and it is now widely used in Response Rate Limiting [78], [79]. Additionally, recent work on DNS privacy solutions is another motivation behind requiring the use of TCP with DNS. One perceived disadvantage associated with DNS over TCP is the higher connection setup latency which is generally equal to one RTT. To amortize connection setup costs and improve efficiency to the maximum, both clients and servers should support connection reuse by sending multiple queries and responses over a single persistent TCP connection.

In practice, the recursive servers widely use cache to improve the resolution efficiency. On one hand, the information of frequently requested names (e.g., root zone file, TLD zone file under the cooperation between the recursive server operator and related registry, top-N names requested by the end users and so on) are cached as many as possible. On the other hand, both positive and negative RRs (e.g., NXDOMAIN, NSEC, NSEC3 and SERVFAIL) are cached for different cases. Besides, many schemes (such as data compression, name redirection, asynchronous communication etc.) are used to improve the DNS efficiency. However, in IoT, the potential name space will grow significantly. To manage this exponential increase in the size of the name space is another big challenge. Besides, the authoritative data is always managed by a cloud server. To enhance scalability and provide name service access from different locations but with a single IP address, DNS makes use of the Border Gateway Protocol (BGP) anycast technology to support a name service at widely distributed geographical locations. In this case, the efficiency of the DNS resolution also depends

on the policy and the performance of the BGP system [80], [81]. Although DNS makes use of a local cache to reduce the resolution latency and anycast to shorten the server distance, it may still not be adequate for the IoT name resolution which requires higher efficiency and scalability.

V. CONCLUSION

The Internet is evolving into the IoT where many devices will be ubiquitously inter-connected as in the future smart city environment. These devices will have different functions and capabilities, but they should be addressed and named somehow. IPv6 supports the addressing of IoT devices with its extended address space, but how to provide a scalable, reliable, and efficient name service (including the name configuration and name resolution) remains a challenge.

In this paper, we analyzed the IoT name service and we have proposed the SMILE requirements based on five characteristics: Security, Mobility, Infrastructure independence, Localization and Efficiency. Next, we focused on DNS and we analyzed the pros and cons of DNS to evaluate its suitability for SMILE. As our analysis has shown, security is enhanced in DNS but the cost is too high for the resource-restricted IoT devices and application scenarios. Although DNS supports dynamic updates but it is not designed for mobility management and the name update is complex. For the name resolution without infrastructure support, DNS can work well with some extensions on the local link and emerging technologies such as edge computing [82]. DNS is also evolving to satisfy the localization requirement for both service deployment and name format. But efficiency remains a significant challenge for DNS because the DNS name resolution mechanism incurs delays due to the hierarchical delegation and unpredictable cache hits.

In summary, DNS has a strong and robust foundation for both enhanced functions and large-scale deployed infrastructures. However, it was not originally designed for IoT applications which will have more strict requirements such as security, mobility, efficiency and so on. Consequently, DNS must be enhanced further to be smarter in order to make it more suitable for SMILE in the emerging ubiquitous IoT world [83], [84]. So, we need to analyze and test the DNS-based name service in more IoT applications so that we can design its functional and performance enhancements according to the special scenario. On the other hand, clean-slate name services [18] should be comprehensively considered within the design frameworks of future Internet architectures which are always motivated by potential IoT applications in the future [85].

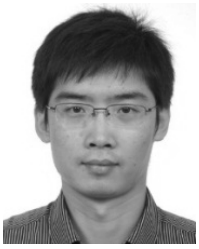
ACKNOWLEDGEMENTS

The authors would like to thank the anonymous reviewers for their valuable comments which helped them to improve the content, organization, and presentation of this paper.

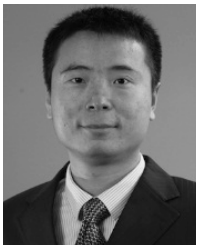
REFERENCES

[1] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," *Comput. Netw.*, vol. 54, no. 15, pp. 2787–2805, Oct. 2010.
[2] P. P. Ray, "A survey on Internet of Things architectures," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 30, no. 3, pp. 291–319, 2018. doi: 10.1016/j.jksuci.2016.10.003.
[3] D. Miorandi, S. Sicari, F. De Pellegrini, and I. Chlamtac, "Internet of Things: Vision, applications and research challenges," *Ad Hoc Netw.*, vol. 10, no. 7, pp. 1497–1516, 2012.
[4] M. D. Lytras and A. Visvizi, "Who uses smart city services and what to make of it: Toward interdisciplinary smart cities research," *Sustainability*, vol. 10, no. 6, p. 1998, 2018. doi: 10.3390/su10061998.
[5] A. Visvizi and M. D. Lytras, "Rescaling and refocusing smart cities research: From mega cities to smart villages," *J. Sci. Technol. Policy Manage.*, vol. 9, no. 2, pp. 134–145, 2018.
[6] A. Visvizi, M. D. Lytras, E. Damiani, and H. Mathkour, "Policy making for smart cities: Innovation and social inclusive economic growth for sustainability," *J. Sci. Technol. Policy Manage.*, vol. 9, no. 2, pp. 126–133, 2018.
[7] B. Kang, D. Kim, and H. Choo, "Internet of everything: A large-scale autonomic IoT gateway," *IEEE Trans. Multi-Scale Comput. Syst.*, vol. 3, no. 3, pp. 206–214, Jul./Sep. 2017.
[8] T. Snyder and G. Byrd, "The Internet of everything," *Computer*, vol. 50, no. 6, pp. 8–9, 2017.
[9] M. H. Miraz, M. Ali, P. S. Excell, and R. Picking, "A review on Internet of Things (IoT), Internet of everything (IoE) and Internet of nano things (IoNT)," in *Proc. Internet Technol. Appl. (ITA)*, Wrexham, U.K., Sep. 2015, pp. 219–224.
[10] M. Sergey, S. Nikolay, and E. Sergey, "Cyber security concept for Internet of Everything (IoE)," in *Proc. Syst. Signal Synchronization, Gener. Process. Telecommun. (SINKHROINFO)*, Kazan, Russia, Jul. 2017, pp. 1–4.
[11] *Gartner*. Accessed: 2019. [Online]. Available: https://www.gartner.com/newsroom/id/3165317
[12] *IEEE Internet of Things*. Accessed: 2015. [Online]. Available: https://iot.ieee.org/newsletter/july-2015/the-case-for-ipv6-as-an-enabler-of-the-internet-of-things.html
[13] T. Gomes, F. Salgado, S. Pinto, J. Cabral, and A. Tavares, "A 6LoWPAN accelerator for Internet of Things endpoint devices," *IEEE Internet Things J.*, vol. 5, no. 1, pp. 371–377, Feb. 2018.
[14] *EPCglobal: EPCglobal Object Name Service (ONS) 1.0.1*, EPCgolbal Standards Develop. Process, New York, NY, USA, 2007.
[15] *HANDLE.NET (Version 8.1) Technical Manual*, CNRI, Reston, VA, USA, 2015.
[16] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher, and B. Ohlman, "A survey of information-centric networking (draft)," in *Proc. Dagstuhl Seminar Inf.-Centric Netw.*, 2011, pp. 1–26.
[17] J. Li, Y. Shvartzshnaider, J.-A. Francisco, R. P. Martin, and D. Raychaudhuri, "Enabling Internet-of-Things services in the mobilityfirst future Internet architecture," in *Proc. World Wireless, Mobile Multimedia Netw. (WoWMoM)*, Jun. 2012, pp. 1–6.
[18] A. M. Alberti, M. A. F. Casaroli, D. Singh, and R. R. Righi, "Naming and name resolution in the future Internet: Introducing the NovaGenesis approach," *Future Gener. Comput. Syst.*, vol. 67, pp. 163–179, Feb. 2017.
[19] A. M. Alberti, M. M. Bontempo, J. R. Santos, A. C. Sodré, and R. Righi, "NovaGenesis applied to information-centric, service-defined, trustable IoT/WSAN control plane and spectrum management," *Sensors*, vol. 18, no. 9, p. 3160, 2018. doi: 10.3390/s18093160.
[20] P. Mockapetris, *Domain Names—Concepts and Facilities*, document RFC 1034, IETF, 1987.
[21] P. Mockapetris and K. J. Dunlap, "Development of the domain name system," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 18, no. 4, pp. 123–133, 1988.
[22] S. Hollenbeck, *Extensible Provisioning Protocol (EPP)*, document RFC 5730, IETF, 2009.
[23] A. Newton, B. Ellacott, and N. Kong, *HTTP Usage in the Registration Data Access Protocol (RDAP)*, document RFC 7480, IETF, 2015.
[24] J. Klensin, *Internationalized Domain Names in Applications (IDNA): Protocol*, document RFC 5891, IETF, 2010.
[25] R. Arends, R. Austein, M. Larson, D. Massey, and S. Rose, *DNS Security Introduction and Requirements*, document RFC 4033, IETF, 2005.
[26] R. Arends, R. Austein, M. Larson, D. Massey, and S. Rose, *Resource Records for the DNS Security Extensions*, document RFC 4034, IETF, 2005.
[27] R. Arends, R. Austein, M. Larson, D. Massey, and S. Rose, *Protocol Modifications for the DNS Security Extensions*, document RFC 4035, IETF, 2005.

[28] P. Hoffman, and J. Schlyter, *The DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS) Protocol: TLSA*, document RFC 6698, IETF, 2012.
[29] *Why is IoT Security important?* Accessed: 2016. [Online]. Available: https://prezi.com/ipid4gsnxvqq/why-is-iot-security-important/
[30] V. Jacobson *et al.*, "Networking named content," in *Proc. ACM CoNEXT*, Rome, Italy, 2009, pp. 1–12.
[31] Y. Zhang *et al.*, *ICN based Architecture for IoT—Requirements and Challenges*, IRTF draft, draft-zhang-iot-icn-challenges-02, 2015.
[32] *Cisco*. Accessed: 2016. [Online]. Available: https://newsroom.cisco.com/press-release-content?articleId=1741352
[33] D. Johnson, C. Perkins, and J. Arkko, *Mobility Support in IPv6*, document RFC 6275, IETF, 2011.
[34] S. Gundavelli, K. Leung, V. Devarapalli, K. Chowdhury, and B. Patil, *Proxy Mobile IPv6*, document RFC 5213, IETF, 2008.
[35] D. Farinacci, V. Fuller, D. Meyer, and D. Lewis, *The Locator/ID Separation Protocol (LISP)*, document RFC 6830, IETF, 2013.
[36] V. P. Kafle, H. Otsuki, and M. Inoue, "An ID/locator split architecture for future networks," *IEEE Commun. Mag.*, vol. 48, no. 2, pp. 138–144, Feb. 2010.
[37] A. Venkataramani *et al.*, "Design requirements of a global name service for a mobility-centric, trustworthy internetwork," in *Proc. IEEE COMSNETS*, Jan. 2013, pp. 1–9.
[38] J. A. Guerrero-Ibáñez, S. Zeadally, and J. Contreras-Castillo, "Integration challenges of intelligent transportation systems with connected vehicle, cloud computing, and Internet of Things technologies," *IEEE Wireless Commun.*, vol. 22, no. 6, pp. 122–128, Dec. 2015.
[39] J. Guerrero-Ibáñez, S. Zeadally, and J. Contreras-Castillo, "Sensor technologies for intelligent transportation systems," *Sensors*, vol. 18, no. 4, p. 1212, 2018. doi: 10.3390/s18041212.
[40] A. Williams, *Requirements for Automatic Configuration of IP Hosts*, IETF draft, draft-ietf-zeroconf-reqts-12, 2002.
[41] F. Siddiqui, S. Zeadally, T. Kacem, and S. Fowler, "Zero configuration networking: Implementation, performance, and security," *Comput. Elect. Eng.*, vol. 38, no. 5, pp. 1129–1145, 2012.
[42] *Functional Architecture*, document ATIS.oneM2M.TS0001V2100-2016, TS 0001, 2016.
[43] J. Contreras-Castillo, S. Zeadally, and J. A. Guerrero-Ibáñez, "Solving vehicular ad hoc network challenges with big data solutions," *IET Netw.*, vol. 5, no. 4, pp. 81–84, 2016.
[44] F. Audet and C. Jennings, *Network Address Translation (NAT) Behavioral Requirements for Unicast UDP*, document RFC 4787, IETF, 2007.
[45] *IETF 6lo WG*. Accessed: 2019. [Online]. Available: https://tools.ietf.org/wg/6lo/
[46] O. Bello, S. Zeadally, and M. Badra, "Network layer inter-operation of device-to-device communication technologies in Internet of Things (IoT)," *Ad Hoc Netw. J.*, vol. 57, pp. 52–62, Mar. 2017.
[47] M. Ain, *D2.3—Architecture Definition, Component Descriptions, and Requirements, Deliverable*, document PSIRP 7th FP EU-funded project, 2009.
[48] Z. Yan, J.-H. Lee, and Y.-J. Park, "Distributed proxies with fast handover support for a PURSUIT based networking architecture," *Wireless Netw.*, vol. 22, no. 1, pp. 307–318, 2016.
[49] S. Ariyapperuma and C. J. Mitchell, "Security vulnerabilities in DNS and DNSSEC," in *Proc. 2nd Int. Conf. Availab., Rel. Secur.*, Apr. 2007, pp. 335–342.
[50] F. Ljunggren, A. M. E. Lowinder, and T. Okubo, *A Framework for DNSSEC Policies and DNSSEC Practice Statements*, document RFC 6841, IETF, 2013.
[51] O. Kolkman, W. Mekking, and R. Gieben, *DNSSEC Operational Practices, Version 2*, document RFC 6871, IETF, 2012.
[52] W. Hardaker, O. Gudmundsson, and S. Krishnaswamy, *DNSSEC Roadblock Avoidance*, document RFC 8027, IETF, 2016.
[53] P. Hoffman and J. Schlyter, *The DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS) Protocol: TLSA*, document RFC 6698, IETF, 2012.
[54] V. Dukhovni and W. Hardaker, *The DNS-Based Authentication of Named Entities (DANE) Protocol: Updates and Operational Guidance*, document RFC 7671, IETF, 2015.
[55] V. Dukhovni and W. Hardaker, *SMTP Security via Opportunistic DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS)*, document RFC 7672, IETF, 2015.
[56] T. Finch, M. Miller, and P. Saint-Andre, *Using DNS-Based Authentication of Named Entities (DANE) TLSA Records with SRV Records*, document RFC 7673, IETF, 2015.
[57] S. Bortzmeyer, *DNS Privacy Considerations*, document RFC 7626, IETF, 2015.
[58] Y. Yao, L. He, and G. Xiong, "Security and cost analyses of DNSSEC protocol," in *Proc. Int. Conf. Trustworthy Comput. Services*, 2012, pp. 429–435.
[59] P. Vixie, S. Thomson, Y. Rekhter, and J. Bound, *Dynamic Updates in the Domain Name System (DNS UPDATE)*, document RFC 2136, IETF, 1997.
[60] B. Wellington, *Secure Domain Name System (DNS) Dynamic Update*, document RFC 3007, IETF, 2000.
[61] *Root Scaling Study: Description of the DNS Root Scaling Model*, TNO Inf. Commun. Technol., New York, NY, USA, 2009.
[62] B. Yahya and J. Ben-Othman, "Achieving host mobility using DNS dynamic updating protocol," in *Proc. 33rd IEEE Conf. Local Comput. Netw. (LCN)*, Oct. 2008, pp. 634–638.
[63] S. Nakajima, "Evaluation of the mobility performance of a personal mobility vehicle for steps," *IEEE Access*, vol. 5, pp. 9748–9756, 2017.
[64] T. Narten, E. Nordmark, W. Simpson, and H. Soliman, *Neighbor Discovery for IP version 6 (IPv6)*, document RFC 4861, IETF, 2007.
[65] R. Droms, *Dynamic Host Configuration Protocol for IPv6 (DHCPv6)*, document RFC 3315, IETF, 2003.
[66] S. Cheshire and M. Krochmal, *DNS-Based Service Discovery*, document RFC 6763, IETF, 2013.
[67] S. Cheshire and M. Krochmal, *Multicast DNS*, document RFC 6762, IETF, 2013.
[68] J. Jeong, *IP Wireless Access in Vehicular Environments (IPWAVE): Problem Statement and Use Cases*, IETF draft, draft-ietf-ipwave-vehicular-networking-07, 2018.
[69] J. Jeong, S. Lee, and J. Park, *DNS Name Autoconfiguration for Internet of Things Devices*, IETF draft, draft-jeong-ipwave-iot-dns-autoconf-04, 2018.
[70] Z. Yan and J.-H. Lee, *Service and Neighbor Discovery in ITS*, IETF Draft, draft-yan-ipwave-nd-03, 2017.
[71] T. Lemon, *Stateful Multi-Link DNS Service Discovery*, IETF draft, draft-lemon-stateful-dnssd-00, 2016.
[72] W. Wang and Z. Yan, *A Survey of the DNS Cache Service in China*, IETF draft, draft-wang-dnsop-cachesurvey-00, 2015.
[73] W. Kumari and P. Hoffman, *Decreasing Access Time to Root Servers by Running One on Loopback*, document RFC 7706, IETF, 2015.
[74] K. Fujiwara, A. Kato, and W. Kumari, *Aggressive Use of DNSSEC-Validated Cache*, document RFC 8198, IETF, 2017.
[75] M. Duerst, *Normalization of Internationalized Identifiers*, IETF draft, draft-duerst-i18n-norm-00, 1997.
[76] J. Klensin, *Internationalized Domain Names in Applications (IDNA): Protocol*, document RFC 5891, IETF, 2010.
[77] J. Dickinson, S. Dickinson, R. Bellis, A. Mankin, and D. Wessels, *DNS Transport Over TCP—Implementation Requirements*, document RFC 7766, IETF, Mar. 2016.
[78] P. Vixie and V. Schryver, *DNS Response Rate Limiting (DNS RRL)*, document ISC-TN 2012-1-Draft1, Apr. 2012.
[79] *Using the Response Rate Limiting Feature in BIND 9.10*, document ISC Support, ISC Knowledge Base AA-00994, Jun. 2013.
[80] J. J. Garcia-Lunes-Aceves, "Loop-free routing using diffusing computations," *IEEE/ACM Trans. Netw.*, vol. 1, no. 1, pp. 130–141, Feb. 1993.
[81] C. Labovitz, A. Ahuja, A. Bose, and F. Jahanian, "Delayed Internet routing convergence," *IEEE/ACM Trans. Netw.*, vol. 9, no. 3, pp. 293–306, Jun. 2001.
[82] S. Wang, X. Zhang, Y. Zhang, L. Wang, J. Yang, and W. Wang, "A survey on mobile edge networks: Convergence of computing caching and communications," *IEEE Access*, vol. 5, pp. 6757–6779, 2017.
[83] V. Pappas, D. Massey, A. Terzis, and L. Zhang, "A comparative study of the DNS design with DHT-based alternatives," in *Proc. IEEE INFOCOM*, Apr. 2006, pp. 1–13.
[84] V. Ramasubramanian and E. G. Sirer, "The design and implementation of a next generation name service for the Internet," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 34, no. 4, pp. 331–342, 2004.
[85] L. F. Herrera-Quintero, J. C. Vega-Alfonso, K. B. A. Banse, and E. C. Zambrano, "Smart ITS sensor for the transportation planning based on IoT approaches using serverless and microservices architecture," *IEEE Intell. Transp. Syst. Mag.*, vol. 10, no. 2, pp. 17–27, Apr. 2018.



ZHIWEI YAN received the Ph.D. degree from the National Engineering Laboratory for Next Generation Internet Interconnection Devices, Beijing Jiaotong University. He joined the China Internet Network Information Center in 2011. Since 2013, he has been an Invited Researcher with Waseda University. He is currently an Associate Professor with the Chinese Academy of Sciences. He is active in IETF and published RFC8191. His research interests include mobility management, network security, and the next generation Internet.



HONGTAO LI received the master's degree in computer science from Tsinghua University, China. As the Project Leader, he carried out the planning and construction of the National Internet Basic Resources Dig Data Service Platform. He is currently the Assistant Director and the Chief Engineer of the China Internet Network Information Center. His research interests include network security and big data technologies. He is a Senior Member of CCF.

SHERALI ZEADALLY received the bachelor's degree in computer science from the University of Cambridge, U.K., and the Ph.D. degree in computer science from the University of Buckingham, U.K. He is currently an Associate Professor with the College of Communication and Information, University of Kentucky. He is a Fellow of the British Computer Society and the Institution of Engineering Technology, U.K.

YU ZENG is currently the Director of the China Internet Network Information Center. He has presided more than 30 key research and development and industrialization projects in the field of information. His research interests include computer architecture, supercomputing, the Internet of Things, and key technologies of Internet fundamental resource management.

GUANGGANG GENG received the Ph.D. degree from the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, China, where he joined the Computer Network Information Center, in 2008. He is currently a Professor with the China Internet Network Information Center. His current research interests include machine learning, adversarial information retrieval on the Web, and Web search.