         DNS Name Autoconfiguration for Internet of Things Devices
                draft-jeong-homenet-device-name-autoconf-04

Abstract

   This document specifies an autoconfiguration scheme for the global
   (or local) DNS names of Internet of Things (IoT) devices, such as
   appliances and sensors. By this scheme, the DNS name of an IoT
   device can be autoconfigured with the device's category and model in
   wired and wireless target networks (e.g., home, office, shopping
   mall, smart grid, and road network). This DNS name lets IoT users
   (e.g., home residents and customers) in the Internet (or local
   network) easily identify each device for monitoring and remote-
   controlling it in the target network.

Table of Contents

1. Introduction

   Many Internet of Things (IoT) devices (e.g., appliances and sensors)
   have begun to have wireless communication capability (e.g., WiFi,
   Bluetooth, and ZigBee) for monitoring and remote-controlling in a
   local network or the Internet. According to the capacity, such IoT
   devices can be categorized into high-capacity devices and low-
   capacity devices. High-capacity devices have a high-power processor
   and a large storage, such as appliances (e.g., television,
   refrigerator, air conditioner, and washing machine) and smart devices
   (smartphone and tablet). They are placed in environments (e.g.,
   home, office, shopping mall, smart grid, and road network) for the
   direct use for human users, and they require the interaction with
   human users. Low-capacity devices have a low-power processor and a
   small storage, such as sensors (e.g., light, meter, room temperature
   controller, and sensors). They are installed for the easy management
   of environments (e.g., home, office, store, and factory), and they do
   not require the interaction with human users.

   For the Internet connectivity of IoT devices, a variety of parameters
   (e.g., address prefixes, default routers, and DNS servers) can be
   automatically configured by Neighbor Discovery (ND) for IP Version 6,
   IPv6 Stateless Address Autoconfiguration, and IPv6 Router
   Advertisement (RA) Options for DNS Configuration [RFC4861][RFC4862]
   [RFC6106].

   For these IoT devices, the manual configuration of DNS names will be
   cumbersome and time-consuming as the number of them increases rapidly
   in a network. It will be good for such DNS names to be automatically
   configured such that they are readable to human users.

   Multicast DNS (mDNS) in [RFC6762] can provide DNS service for
   networked devices on a local link (e.g., home network and office
   network) without any conventional recursive DNS server. mDNS also
   supports the autoconfiguration of a device's DNS name without the
   intervention of the user. mDNS aims at the DNS naming service for the
   local DNS names of the networked devices on the local link rather
   than the DNS naming service for the global DNS names of such devices
   in the Internet. However, for IoT devices accessible from the
   Internet, mDNS cannot be used. Thus, a new autoconfiguration scheme
   becomes required for the global DNS names of IoT devices.

   This document proposes an autoconfiguration scheme for the global (or
   local) DNS names of IoT devices. Since an autoconfigured DNS name
   contains the device category and model of a device, IoT users in the
   Internet (or local network) can easily identify the device. With
   this device category and model, they will be able to monitor and
   remote-control each device with mobile smart devices (e.g.,

smartphone and tablet) by resolving its DNS name into the
corresponding IPv6 address.

1.1. Applicability Statements

It is assumed that IoT devices have networking capability through
wired or wireless communication media, such as Ethernet [IEEE-802.3],
WiFi [IEEE-802.11] [IEEE-802.11a] [IEEE-802.11b][IEEE-802.11g]
[IEEE-802.11n], Bluetooth [IEEE-802.15.1], and ZigBee [IEEE-802.15.4]
in a local area network (LAN) or personal area network (PAN).

Also, it is assumed that each IoT device has a factory configuration
(called device configuration) having device category (e.g., smart TV,
smartphone, tablet, and refrigerator) and model (i.e., a specific
model name of the device). This device configuration can be read by
the device for DNS name autoconfiguration.

2. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
"SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this
document are to be interpreted as described in RFC 2119 [RFC2119].

3. Terminology

This document uses the terminology described in [RFC4861] and
[RFC4862]. In addition, four new terms are defined below:

o Device Configuration: A factory configuration that has device
   category (e.g., smart TV, smartphone, tablet, and refrigerator)
   and model (i.e., a specific model name of the device).

o DNS Search List (DNSSL): The list of DNS suffix domain names used
   by IPv6 hosts when they perform DNS query searches for short,
   unqualified domain names [RFC6106].

o DNSSL Option: IPv6 RA option to deliver the DNSSL information to
   IPv6 hosts [RFC6106].

4. Overview

This document specifies an autoconfiguration scheme for an IoT device
using device configuration and DNS search list. Device configuration
has device category and device model. DNS search list has DNS suffix
domain names that represent the DNS domains of a network having the
IoT device [RFC6106].

As an IPv6 host, the IoT device can obtain DNS search list through

IPv6 Router Advertisement (RA) with DNS Search List (DNSSL) Option
[RFC4861][RFC6106] or DHCPv6 with Domain Search List Option
[RFC3315][RFC3736][RFC3646].

The IoT device can construct its DNS name with the concatenation of
device category, device model, and domain name. Since there exist
more than one device with the same model, the DNS name should have a
unique identification to differentiate multiple devices with the same
model.

Since both RA and DHCPv6 can be simultaneously used for the parameter
configuration for IPv6 hosts, this document considers the DNS name
autoconfiguration in the coexistence of RA and DHCP.

5. DNS Name Autoconfiguration

The DNS name autoconfiguration for an IoT device needs the
acquisition of DNS search list through either RA [RFC6106] or DHCPv6
[RFC3646]. Once the DNS search list is obtained, the IoT device
autonomously constructs its DNS name(s) with the DNS search list and
its device information.

5.1. DNS Name Format

A DNS name for an IoT device has the following format as in Figure 1:

```
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |    unique_id.device_model.device_category.domain_name    |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

                  Figure 1: IoT Device DNS Name Format

Fields:

  unique_id   unique identifier to guarantee the uniqueness
                     of the DNS name in ASCII characters. The
                     identifier MAY be a sequence number or
                     alphanumeric with readability, such as product
                     name.

  device_model   device's model name in ASCII characters. It
                     is a product model name provided by the
                     manufacturer.

  device_category   device's category name in ASCII characters,
                     such as TV, refrigerator, air conditioner,
                     smartphone, tablet, light, and meter.

domain_name   DNS domain name that is encoded according to
                     the specification of "Representation and use
                     of domain name" of RFC 3315.

5.2. Procedure of DNS Name Autoconfiguration

The procedure of DNS name autoconfiguration is performed through a
DNSSL option delivered by either RA [RFC6106] or DHCPv6 [RFC3646].

5.2.1. DNS Name Generation

When as an IPv6 host a device receives a DNSSL option through either
RA or DHCPv6, it checks the validity of the DNSSL option. If the
option is valid, the IPv6 host performs the DNS name
autoconfiguration with each DNS suffix domain name in the DNSSL
option as follows:

1. The host constructs its DNS name with the DNS suffix domain name
    along with device configuration (i.e., device category and device
    model) and a selected identifier (as unique_id) that is
    considered unique, as shown in Figure 1.

2. The host constructs an IPv6 unicast address as a tentative
    address with a 64-bit network prefix and the last 64 bits of the
    MD5 hashed value of the above DNS name.

3. The host constructs the solicited-node multicast address in
    [RFC4861] corresponding to the tentative IPv6 address.

4. The host performs Duplicate Address Detection (DAD) for the IPv6
    address with the solicited-node multicast address [RFC4861]
    [RFC4862].

5. If there is no response from the DAD, the host sets the IPv6
    tentative address as its IPv6 unicast address and regards the
    constructed DNS name as unique on the local link. Otherwise,
    since the DAD fails because of DNS name conflict, go to Step 1
    for a new DNS name generation with another identifier for
    unique_id.

6. Since the DNS name is proven to be unique, it is used as the
    device's DNS name and the DNS autoconfiguration is done for the
    given DNS suffix domain name. Also, the host joins the
    solicited-node multicast address for the verified DNS name in
    order to prevent other hosts from using this DNS name.

When the DNS search list has more than one DNS suffix domain name,
the IPv6 host repeats the above procedure until all of the DNS

suffixes are used for the DNS name autoconfiguration along with the
IPv6 unicast autoconfiguration corresponding to the DNS name.

5.2.2. DNS Name Collection

Once as IPv6 hosts the devices have autoconfigured their DNS names,
as a collector, any IPv6 node (i.e., router or host) in the same
subnet can collect the device DNS names using IPv6 Node Information
(NI) protocol [RFC4620].

For a collector to collect the device DNS names without any prior
node information, a new NI query needs to be defined. That is, a new
ICMPv6 Code (e.g., 3) SHOULD be defined for the collection of the
IPv6 host DNS names. The Data field is not included in the ICMPv6
header since the NI query is for all the IPv6 hosts in the same
subnet. The Qtype field for NI type is set to 2 for Node Name.

The query SHOULD be transmitted by the collector to a link-local
multicast address for this NI query. Assume that a link-local scope
multicast address (e.g., all-nodes multicast address, FF02::1) SHOULD
be defined for device DNS name collection such that all the IPv6
hosts join this link-local multicast address for the device DNS name
collection service.

When an IPv6 host receives this query sent by the collector in
multicast, it transmits its Reply with its DNS name with a random
interval between zero and Query Response Interval, as defined by
Multicast Listener Discovery Version 2 [RFC3810]. This randomly
delayed Reply allows the collector to collect the device DNS names
with less frame collision probability by spreading out the Reply time
instants.

After the collector collects the device DNS names, it resolves the
DNS names into the corresponding IPv6 addresses by NI protocol
[RFC4620] with the ICMPv6 Code 1 of NI Query. This code indicates
that the Data field of the NI Query has the DNS name of an IoT
device. The IoT device that receives this NI query sends the
collector an NI Reply with its IPv6 address in the Data field.

For DNS name resolution service, the collector can register the
pair(s) of DNS name and IPv6 address for each IPv6 host into an
appropriate designated DNS server for the DNS domain suffix of the
DNS name. It is assumed that the collector is configured to register
DNS names into the designated DNS server in a secure way based on
DNSSEC [RFC4033][RFC6840]. This registration of the DNS name and
IPv6 address can be performed by DNS dynamic update [RFC2136].
Before registering the DNS name into the designated DNS server, the
collector SHOULD verify the uniqueness of the DNS name in the

intended DNS domain by sending a DNS query for the resolution of the
DNS name. If there is no corresponding IPv6 address for the queried
DNS name, the collector registers the DNS name and the corresponding
IPv6 address into the designated DNS server. On the other hand, if
there is such a corresponding IPv6 address, the DNS name is regarded
as duplicate (i.e., not unique), and so the corresponder notifies the
corresponding IoT device with the duplicate DNS name of an error
message of DNS name duplication using NI protocol. When an IoT
device receives such a DNS name duplication error, it needs to
construct a new DNS name and repeats the procedure of device DNS name
generation along with the uniqueness test of the device DNS name in
its subnet.

The two separate procedures of the DNS name collection and IPv6
address resolution in the above NI protocol can be consolidated into
a single collection for the pairs of DNS names and the corresponding
IPv6 addresses. For such an optimization, a new ICMPv6 Code (e.g.,
4) is defined for the NI Query to query the pair of a DNS name and
the corresponding IPv6 address. With this code, the collector can
collect the pairs of each IoT device's DNS name and IPv6 address in
one NI query message rather than two NI query messages.

5.2.3. DNS Name Retrieval

A smart device like smartphone can retrieve the DNS names of IoT
devices by contacting a global (or local) DNS server having the IoT
device DNS names. If the smart device can retrieve the zone file
with the DNS names, it can display the information of IoT devices in
a target network, such as home network and office network. With this
information, the user can monitor and control the IoT devices in the
Internet (or local network).

6. Location-Aware DNS Name Configuration

If the DNS name of an IoT device includes location information, it
allows users to easily identify the physical location of each device.
This document proposes the representation of location in a DNS name.

6.1. Macro-Location-Aware DNS Name

If location information (such as living room, kitchen, and bedroom in
an apartment) is available to an IoT device, a keyword for the
location can be used to construct a DNS name as subdomain name. This
location information lets users track the position of mobile devices
(such as smartphone, tablet, and vacuum cleaning robot). The
physical location of the device is defined as macro-location for DNS
naming.

A subdomain name for macro-location (denoted as mac_loc) MAY be
placed between device_category and domain_name of the DNS name format
in Figure 2. A localization scheme for device location is beyond the
scope of this document.

6.2. Micro-Location-Aware DNS Name

An IoT device can be located in the center, wall, or corner in a room
that is specified by macro-location. For example, assume that a
cleaning robot is located in the right-upper corner of a living room.
If the DNS name for the cleaning robot contains the right-upper
corner of the living room, a home resident can find it easily. In
this document, for this DNS naming, the detailed location for an IoT
device can be specified as a micro-location subdomain name.

A subdomain name for micro-location (denoted as mic_loc) MAY be
placed between device_category and domain_name of the DNS name format
in Figure 2. A localization scheme for micro-location is beyond the
scope of this document.

To denote both macro-location (i.e., mac_loc) and micro-location
(i.e., mic_loc) into a DNS name, the following format is described as
in Figure 2:

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| unique_id.device_model.device_category.mic_loc.mac_loc.domain_name|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 2: Location-Aware Device DNS Name Format

Fields:

  unique_id   unique identifier to guarantee the uniqueness
                     of the DNS name in ASCII characters. The
                     identifier MAY be a sequence number or
                     alphanumeric with readability, such as product
                     name.

  device_model   device's model name in ASCII characters. It
                     is a product model name provided by the
                     manufacturer.

  device_category   device's category name in ASCII characters,
                     such as TV, refrigerator, air conditioner,
                     smartphone, tablet, light, and meter.

  mic_loc   device's micro-location, such as center,
                     wall, and corner.

    mac_loc   device's macro-location, such as living room.

    domain_name   DNS domain name that is encoded according to
                         the specification of "Representation and use
                         of domain name" of RFC 3315.

7. DNS Name Management for Mobile IoT Devices

   Some IoT devices can have mobility, such as smartphone, tablet,
   laptop computer, and cleaning robot. This mobility allows the IoT
   devices to move from a subnet to another subnet where subnets can
   have different domain suffixes, such as living_room.home and
   garage.home. The DNS name change (or addition) due to the mobility
   should be considered.

   To deal with DNS name management in mobile environments, whenever an
   IoT device enters a new subnet and receives DNS suffix domain names,
   it generates its new DNS names and registers them into a designated
   DNS server, specified by RDNSS option.

   When the IoT device recognizes the movement to another subnet, it can
   delete its previous DNS name(s) from the DNS server having the DNS
   name(s), using DNS dynamic update [RFC2136]. For at least one DNS
   name to remain in a DNS server for the location management in Mobile
   IPv6 [RFC6275], the IoT device does not delete its default DNS name
   in its home network in Mobile IPv6.

8. Security Considerations

   This document shares all the security issues of the NI protocol that
   are specified in the "Security Considerations" section of [RFC4620].

   To prevent the disclosure of location information for privacy
   concern, the subdomains related to location can be encrypted by a
   shared key or public-and-private keys. For example, a DNS name of
   smartphone1.living_room.home can be represented as
   smartphone1.xxx.home where xxx is a string of the encrypted
   representation of the subdomain living_room.

9. Acknowledgements

   This work was partly supported by the ICT R&D program of MSIP/IITP
   [10041244, SmartTV 2.0 Software Platform] and ETRI.

10. References

10.1. Normative References

[RFC2119]   Bradner, S., "Key words for use in RFCs to Indicate
                 Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC4861]   Narten, T., Nordmark, E., Simpson, W., and H.
                 Soliman, "Neighbor Discovery for IP Version 6
                 (IPv6)", RFC 4861, September 2007.

[RFC4862]   Thomson, S., Narten, T., and T. Jinmei, "IPv6
                 Stateless Address Autoconfiguration", RFC 4862,
                 September 2007.

[RFC6106]   Jeong, J., Park, S., Beloeil, L., and S.
                 Madanapalli, "IPv6 Router Advertisement Options for
                 DNS Configuration", RFC 6106, November 2010.

[RFC3315]   Droms, R., Ed., Bound, J., Volz, B., Lemon, T.,
                 Perkins, C., and M. Carney, "Dynamic Host
                 Configuration Protocol for IPv6 (DHCPv6)", RFC 3315,
                 July 2003.

[RFC3736]   Droms, R., "Stateless Dynamic Host Configuration
                 Protocol (DHCP) Service for IPv6", RFC 3736,
                 April 2004.

[RFC3646]   Droms, R., Ed., "DNS Configuration options for
                 Dynamic Host Configuration Protocol for IPv6
                 (DHCPv6)", RFC 3646, December 2003.

[RFC4033]   Arends, R., Ed., Austein, R., Larson, M., Massey,
                 D., and S. Rose, "DNS Security Introduction and
                 Requirements", RFC 4033, March 2005.

[RFC6840]   Weiler, S., Ed. and D. Blacka, Ed., "Clarifications
                 and Implementation Notes for DNS Security (DNSSEC)",
                 RFC 6840, February 2013.

10.2. Informative References

[RFC6762]   Cheshire, S. and M. Krochmal, "Multicast DNS",
                 RFC 6762, February 2013.

[RFC4620]   Crawford, M. and B. Haberman, Ed., "IPv6 Node
                 Information Queries", RFC 4620, August 2006.

[RFC3810]   Vida, R. and L. Costa, "Multicast Listener Discovery
                 Version 2 (MLDv2) for IPv6", RFC 3810, June 2004.

    [RFC2136]   Vixie, P., Ed., Thomson, S., Rekhter, Y., and J.
                       Bound, "Dynamic Updates in the Domain Name System
                       (DNS UPDATE)", RFC 2136, April 1997.

    [RFC6275]   Perkins, C., Ed., Johnson, D., and J. Arkko,
                       "Mobility Support in IPv6", RFC 6275, July 2011.

    [IEEE-802.3]   IEEE Std 802.3, "IEEE Standard for Ethernet",
                       December 2012.

    [IEEE-802.11]   IEEE Std 802.11, "Part 11: Wireless LAN Medium
                       Access Control (MAC) and Physical Layer (PHY)
                       Specifications", March 2012.

    [IEEE-802.11a]   IEEE Std 802.11a, "Part 11: Wireless LAN Medium
                       Access Control (MAC) and Physical Layer (PHY)
                       specifications: High-speed Physical Layer in the 5
                       GHZ Band", September 1999.

    [IEEE-802.11b]   IEEE Std 802.11b, "Part 11: Wireless LAN Medium
                       Access Control (MAC) and Physical Layer (PHY)
                       specifications: Higher-Speed Physical Layer
                       Extension in the 2.4 GHz Band", September 1999.

    [IEEE-802.11g]   IEEE P802.11g/D8.2, "Part 11: Wireless LAN Medium
                       Access Control (MAC) and Physical Layer (PHY)
                       specifications: Further Higher Data Rate Extension
                       in the 2.4 GHz Band", April 2003.

    [IEEE-802.11n]   IEEE P802.11n/D9.0, "Part 11: Wireless LAN Medium
                       Access Control (MAC) and Physical Layer (PHY)
                       specifications Amendment 5: Enhancements for Higher
                       Throughput", March 2009.

    [IEEE-802.15.1]   IEEE Std 802.15.1, "Part 15.1: Wireless Medium
                       Access Control (MAC) and Physical Layer (PHY)
                       specifications for Wireless Personal Area Networks
                       (WPANs)", June 2005.

    [IEEE-802.15.4]   IEEE Std 802.15.4, "Part 15.4: Low-Rate Wireless
                       Personal Area Networks (LR-WPANs)", September 2011.

Authors' Addresses

    Jaehoon Paul Jeong
    Department of Software
    Sungkyunkwan University
    2066 Seobu-Ro, Jangan-Gu
    Suwon, Gyeonggi-Do 440-746
    Republic of Korea

    Phone: +82 31 299 4957
    Fax: +82 31 290 7996
    EMail: pauljeong@skku.edu
    URI: http://cpslab.skku.edu/people-jaehoon-jeong.php

    Sejun Lee
    Department of Computer Science and Engineering
    Sungkyunkwan University
    2066 Seobu-Ro, Jangan-Gu
    Suwon, Gyeonggi-Do 440-746
    Republic of Korea

    Phone: +82 31 299 4106
    Fax: +82 31 290 7996
    EMail: sejunlee@skku.edu

    Jung-Soo Park
    Electronics and Telecommunications Research Institute
    218 Gajeong-Ro, Yuseong-Gu
    Daejeon,   305-700
    Republic of Korea

    Phone: +82 42 860 6514
    EMail: pjs@etri.re.kr