|  |
| --- |
|  |

|  |
| --- |
| Input Contribution |
| Meeting ID\* | RDM#59 |
| Title:\* | DNS-SD based oneM2M device discovery and registration |
| Source:\* | JaeSeung Song, Sejong University & KETI, jssong@sejong.ac.kr JiEun Lee, Sejong University, love9ly@gmail.comJiHo Lee, Sejong University, l22twozio@gmail.com |
| Date:\* | 2023-04-18 |
| Input related to\* | TR-0059 (Rel-5) |
| Intended purpose ofdocument:\* | [x]  Decision[ ]  Discussion[ ]  Information[ ]  Other <specify> |
| Impacted other TS/TR(s) |  |
| Decision requested or recommendation:\* | Agree for inclusion in TR-0059 Services and platforms discovery |
| Template Version: January 2017 (Do not modify) |

**oneM2M Notice**

The document to which this cover statement is attached is submitted to oneM2M. Participation in, or attendance at, any activity of oneM2M, constitutes acceptance of and agreement to be bound by terms of the Working Procedures and the Partnership Agreement, including the Intellectual Property Rights (IPR) Principles Governing oneM2M Work found in Annex 1 of the Partnership Agreement.

# Introduction

This contribution introduces a new proposed concept to support DNS-SD based oneM2M device discovery and registration.

### -----------------------Start of change 1-------------------------------------------

7 Concepts for Service and Platform Discovery

7.1 DNS-SD based oneM2M Service Discovery

7.1.1 Overview

To enable a oneM2M entity to query and discover available oneM2M SPs and their available services, this concept proposes to leverage DNS-SD[i.2]. This concept is ideal for the scenario where a oneM2M entity is not aware of the network addresses of available oneM2M nodes in the system, the oneM2M CSEs and AEs hosted upon these nodes, and when applicable, the oneM2M SPs that manages these services.

7.1.2 Conceptual Description

DNS-SD can be used by a oneM2M entity to find M2M SPs, the oneM2M nodes that they deploy and the CSEs and AEs deployed on these nodes. DNS-SD is well-suited for cases when a oneM2M entity is not pre-provisioned with this information and needs to discover it in a more ad-hoc/dynamic fashion. For example, an ADN-AE that wants to search for a new type of service. In such cases, a oneM2M entity can query a DNS-SD server to discover a particular type of service, the M2M SP that deploys the service and the oneM2M node(s) that host CSE(s) or AE(s) that offer the service.

NOTE: DNS-SD relies on awareness of the network address of the DNS-SD server. This address can be provisioned onto a oneM2M node/entity. For example, a DNS-SD client hosted on a ADN/MN/ASN can be configured during the process of enrolling to the MEF with the network address of one or more DNS-SD servers operated by a M2M SP. Alternatively, the network address of one or more DNS-SD servers can be configured when bootstrapping with an underlying access network that is managed by a network operator.

A DNS-SD server can be deployed in a oneM2M System as shown in Figure 7.1.2-1. oneM2M entities wishing to perform discovery of oneM2M services can support a DNS-SD client. The DNS-SD client can be co-hosted on the same node as a oneM2M entity or integrated within the oneM2M entity. Detailed example procedure for the DNS-SD based service discovery is described in Annex <A>.



**Figure 7.1.2-1: DNS-SD Server deployed in oneM2M System**

* 1. oneM2M CSEs Discovery

7.2.1 Overview

To enable a oneM2M entity to discover available oneM2M CSEs based on certain criteria (such as location, supporting services), this concept proposes to introduce a Registry server that stores oneM2M CSE discovery information and that can be queried by entities in the oneM2M System. This concept is ideal for the case where multiple oneM2M CSEs are available in a smart city or state or a country, so that a oneM2M application can select a required oneM2M CSEs even it does not have any prior information about available oneM2M CSEs. This concept can guarantee that only properly working oneM2M CSEs are discoverable based on their availability status.

* + 1. Conceptual Description using Registry

This clause describes a high-level concept for a centralized CSE discovery registry-based. In order to discover available oneM2M CSEs, a service registry that stores information regarding available oneM2M CSEs is required. A proper description of oneM2M CSEs and registration procedures are needed to be defined.

 

**Figure 7.2.2-1: High-level concept for a Registry based oneM2M**

**CSE discovery**

CSEs discovery is the process of locating CSE information and retrieving CSE descriptions that have been previously stored within the Registry. oneM2M AEs query the Registry for information regarding available oneM2M CSEs matching the requirements of the AEs. Examples of CSE information that can be stored in the Registry can include information the following:

* Contact of Address: IP Address of a CSE
* Port number: Port number for a CSE
* Name: Name of a CSE
* Status: Liveness information (whether a CSE is running or not)
* Location: Geography information where a CSE is located in.
* Profile: Information about which product profile is referred by a CSE
* Type of CSE (MN-CSE or IN-CSE)
* Supporting public services (e.g., smart parking, smart home)
* Maintenance information (for example, from 01:00 ~ 02:00)
* Access information (or credentials)
* Physical capability (e.g., available storage, memory, etc.)

After the CSE discovery procedure is completed, the oneM2M AE can know the exact location of a needed oneM2M CSE via PoA, its capabilities, and how to communicate with it.

The registry for oneM2M CSEs is providing a similar service to the oneM2M App-ID registry. The PoA of the central Registry can be pre-provisioned.

As there exist oneM2M CSEs which are not available because of various reasons such as maintenance, out of order and temporary disorder, the registry has to check the availability status, which refers to the fact that a CSE is running or not, of the registered oneM2M CSEs periodically.

Further detailed example procedure is described in Annex <B>.

7.3 DNS-SD based oneM2M Device Discovery and Registration

7.3.1 Overview

Registering and using newly purchased IoT devices in an existing IoT service can be a hassle due to complicated settings and manual work. For instance, the user is required to perform detailed and complex tasks, such as manually setting a network function for each IoT device to be serviced and linking it with a server. To register an IoT device in the existing oneM2M system, the user must individually utilize the IP address and resource information of the oneM2M platform for each device to complete the registration process. This process involves manually including various information, such as the ID, type, and name of each device in the registration information, and transmitting it. Additionally, the existing oneM2M discovery function only allows to find out devices that have already been registered, and there is no search function for a new IoT device to be registered, causing inconvenience. However, in the case of the Zeroconf network protocol, devices and services can be dynamically searched for and registered in a distributed environment. Therefore, if this function can be provided in the oneM2M IoT platform, IoT devices that can be registered automatically are discovered without manual information input by the user. The selected IoT device can be automatically registered and used on the server platform.

7.3.2 Description of DNS-SD based oneM2M Device Discovery

The DNS-SD based oneM2M device discovery function provides an easier way to register oneM2M devices by applying Zeroconf's DNS-SD function, which enables automatic IT service and device discovery, to the oneM2M server platform and devices. As shown in Figure 8.3.2-1, in order to enable these functions, the following parts need to be applied to existing oneM2M entities:

* oneM2M server platform: A function to discover oneM2M type IoT devices using the DNS-SD protocol and a function to use the selected device information to register oneM2M platform.
* oneM2M device: It needs a function that can multicast its own type as a oneM2M device type and the basic information necessary for oneM2M platform registration using the DNS-SD protocol.
* oneM2M device registration application: It includes a function to request a oneM2M type device search from the oneM2M platform and request oneM2M platform registration for a specific selected device.



Figure 8.3.2-1 A high-level concept of DNS-SD based oneM2M device discovery and registration

Basically, the user queries the oneM2M platform through the oneM2M application to check if there are any new oneM2M devices that can be registered on the current network. Upon receiving the request, the oneM2M platform sends a DNS-SD multicast message to check if there are any oneM2M type devices and services that can be registered on the network using the DNS-SD function. When oneM2M devices with DNS-SD function receive a message from the IoT platform that they are searching for a oneM2M type device, they respond to it. The oneM2M platform transfers the collected device information that sent the response to the oneM2M application. The user looks at a list of available oneM2M devices, selects a device to be registered, and sends a registration request for the selected device to the oneM2M platform. The oneM2M platform then creates a resource by performing a registration procedure for the selected device and makes it available to the oneM2M application.

The detailed example procedure related to introduced DNS-SD based oneM2M type device discovery and registration is described in Annex <C>.

# Conclusions

*Editor’s Note: This section provides a summary of the conclusions drawn during the study.*

Annex <A>: Example Procedure for DNS-SD based Service Discovery

Section 7.1 describes the concept of DNS-SD based oneM2M service discovery. The following procedure describes the steps involved in the concept of DNS-SD based oneM2M service discovery.



**Figure Annex A-1: DNS-SD based oneM2M Service Discovery**

**Step 1:** *(Offline and Optional)* If a private DNS-SD deployment is preferred, then this step can be skipped. The DNS-SD server is registered with an official DNS registrar company/entity to establish a public DNS-SD service discovery domain. The DNS registrar then makes this information publically available. Other DNS-SD servers can then discover any new service discovery sub-domains that have been established under their level of hierarchy. This results in a M2M SP’s DNS-SD server being discovered and interconnected into the hierarchy of existing DNS-SD servers.

**Step 2:** *(Offline and Optional)* Each M2M Service Provider provisions Service Provider (SP) discovery information (DNS-SD PTR, SRV and TXT records) in their DNS-SD servers. These discovery records include information regarding the different types of services offered, locations that these services are offered, terms and conditions for accessing the services, point of access information for SP functions such as MEF and MAF functions, etc.

 Editor’s Note: Definition of standardized service types for use in DNS-SD based oneM2M SP discovery is FFS

**Step 3:** *(Offline)* Each M2M Service Provider provisions CSE discovery information (DNS-SD PTR, SRV and TXT records) in the DNS-SD server. These discovery records include information describing each CSE instance the SP has deployed. The information includes the network domain or location where the CSE is deployed, the types of services (e.g. CSFs) supported by the CSE, point of access information for a CSE, supported protocol bindings and content serialization formats, etc.

Editor’s Note: Definition of standardized service types for use in DNS-SD based oneM2M CSE discovery is FFS

**Step 4:** *(Offline and Optional)* Each M2M Service Provider can provision AE discovery information (DNS-SD PTR, SRV and TXT records) in the DNS-SD server. These discovery records include information describing each AE instance associated with the SP. The information includes the network domain or location where the AE is deployed, the types of services supported by the AE, which CSE the AE is accessible via, supported information models and content serialization formats, etc.

Editor’s Note: Definition of standardized service types for use in DNS-SD based oneM2M AE discovery is FFS

**Step 5:** The Registree Node’s DNS-SD client either discovers, is configured, or is provisioned with the identity of its local DNS-SD server. Depending on the underlying access network technology, this process may be automated or it may require manual offline configuration. Standardized DNS-SD service types for oneM2M SPs, CSEs and/or AEs are also configured such that DNS-SD based discovery lookups can be conducted by the Registree.

**Step 6:** *(Optional)* The Registree AE/CSE initiates a M2M SP discovery procedure by sending a DNS PTR query (that includes a standardized and/or registered oneM2M SP type and the domain) to the DNS-SD server.

**Step 7:** *(Optional)* The DNS-SD server responds to the Registree AE/CSE with a list of available DNS-SD PTR record(s), each containing a M2M SP ID of a corresponding SP.

**Step 8:** *(Optional)* The Registree AE/CSE selects one or more PTR records that it would like to resolve in order to get additional discovery information for the SP(s) of interest.

**Step 9:** *(Optional)* The Registree AE/CSE sends DNS query request(s) to the DNS-SD server for each SP it would like to lookup additional information about. Note, separate requests are needed for each SP.

**Step 10:** *(Optional)* The DNS-SD server responds with the SRV and TXT records for each corresponding DNS-SD lookup request. The SRV record contains the network address of the SP’s enrolment function (MEF) (e.g. IP address and port). The TXT record contains additional information such as the different types of services the SP offers, locations that these services are offered, terms and conditions for accessing the services, etc.

**Step 11:** *(Optional)* Using information from the TXT record to find the most suitable SP, the Registree AE/CSE selects SP(s) to enrol to.

**Step 12:** The Registree AE/CSE initiates a M2M CSE discovery procedure by sending a DNS PTR query (that includes a standardized and/or registered oneM2M CSE type and the domain) to the DNS-SD server.

**Step 13:** The DNS-SD server responds to the Registree AE/CSE with a list of available DNS-SD PTR record(s), each containing a M2M CSE ID of a corresponding CSE.

**Step 14:** The Registree AE/CSE selects one or more PTR records that it would like to resolve in order to get additional discovery information for the CSE(s).

**Step 15:** The Registree AE/CSE sends DNS query request(s) to the DNS-SD server for each CSE it would like to lookup additional information about. Note, separate requests are needed for each CSE.

**Step 16:** The DNS-SD server responds with the SRV and TXT records for each corresponding DNS-SD lookup request. The SRV record contains the network address of the CSE (e.g. IP address and port). The TXT record contains additional information such as the URI path to <CSEBase>, types/classes of services supported by the CSE, supported protocol bindings and serialization types, etc. supported by the CSE.

**Step 17:** Using information from the TXT record to find the most suitable CSE(s), the Registree AE/CSE selects CSE(s) to perform a security association with and also register to.

**Step 18:** *(Optional)* The Registree AE/CSE initiates a M2M AE discovery procedure by sending a DNS PTR query (that includes a standardized and/or registered oneM2M AE type (e.g. App-ID) and the domain) to the DNS-SD server.

**Step 19:** *(Optional)* The DNS-SD server responds to the Registree AE/CSE with a list of available DNS-SD PTR record(s), each containing oneM2M AE information (e.g. AE-ID and/or App-ID) of a corresponding AE.

**Step 20:** *(Optional)* The Registree AE/CSE selects one or more PTR records that it would like to resolve in order to get additional discovery information for the AE.

**Step 21:** *(Optional)* The Registree AE/CSE sends DNS query request(s) to the DNS-SD server for each AE it would like to lookup additional information about. Note, separate requests are needed for each AE.

**Step 22:** *(Optional)* The DNS-SD server responds with the SRV and TXT records for each corresponding DNS-SD lookup request. The SRV record contains the network address of the Registrar CSE of the AE (e.g. IP address and port). The TXT record contains additional information about the AE such as the URI path to <AE> resource hosted on the Registrar CSE, types/classes of services supported by the AE, supported information model and serialization types, etc.

**Step 23:** *(Optional)* Using information from the TXT record to find the most suitable AE(s), the Registree AE/CSE selects AE(s) to interact with.

Annex <B>: Example Procedure for oneM2M CSE Discovery

Section 7.2 describes the concept of oneM2M CSE discovery. The following procedure describes the steps involved in the concept of oneM2M CSE discovery.

This Annex especially describes a mechanism for oneM2M CSE discovery based on a dedicated resource named <*cseRegistry*> and <*cseRegistryList*>. In order to discover oneM2M CSEs, the <*cseRegistry*> resource can be used to keep all the available oneM2M CSEs and their description to be discovered either open to the public or have a business relationship. <*cseRegistryList*> resource is used to manage <*cseRegistry*> as shown in Figure Annex B-1. For example, if a citizen from smart city A visits smart city B, the citizen may want to find out available parking lots using oneM2M smart parking application. In this case, the oneM2M smart parking application can discover available CSEs supporting smart parking service in smart city B via looking into the <*cseRegistry*> resource. As another example, an oneM2M application running on a moving vehicle tries to find out available CSEs that are covering the routes where the vehicle is going to take and supporting Edge/Fog capability.

 

Figure Annex B-1: Resource structure showing how <*cseRegistry*> and <*cseRegistryList*> are composed of

In this case, synchronization between the attributes of remote CSEs in the <*cseRegistry*> resource and the original resource (i.e., available oneM2M CSEs) shall be the responsibility of the <*cseRegistry*> resource hosting CSE. In addition, the hosting CSE performs the liveness check for the all the remoteCSEs managed in the <*cseRegistry*> resource.

An AE or other CSE can request the source CSE for adding its information to the <*caseRegistry*> resource of the target Hosing CSE.

**Step 001**: The Originator of a Request initiating the publication of the description of a CSE to a target CSE. The request should include the target CSE address and the indication of publishing CSE description. The target CSE can also be the Originator of a Request.

**Step 002**: The Hosting CSE then prepares a CREATE request message of itself to the given target CSE’s <*cseRegistry*> resource. The message is composed of the address of itself, access token to be used for basic authentication, supporting services and features.

**Step 003**: The Hosting CSE sends the CREATE request message to the target CSE

**Step 004**: The Target CSE adds a new CSE description to the <*cseRegistry*> resource

 

Figure Annex B-2: Procedure for creating a new CSE description record to the <*cseRegistry*> resource

Annex <C>: Example Procedure for DNS-SD based oneM2M Device Discovery and Registration

Section 7.3 describes the concept of DNS-SD based oneM2M device discovery and registration. The following procedure describes the detailed example steps to support oneM2M device discovery and registration.



Figure Annex C-1: DNS-SD based oneM2M Service Discovery

**Step 01:** Assume there are unregistered oneM2M IoT devices that use the DNS-SD function to advertise their availability on the local network by continuously broadcasting a message indicating that they are oneM2M type devices at regular intervals.

**Step 02:** A user who wants to register a new oneM2M device sends a request to the oneM2M platform to search for a oneM2M type device using the oneM2M application.

**Step 03:** The oneM2M platform uses the DNS-SD function to transmit a DNS-SD query to the local network to search for oneM2M type devices based on the received request.

**Steps 04 ~ 05:** The corresponding oneM2M type devices receiving the query respond to the oneM2M platform by sending a response message that includes their information, including device information necessary for registration.

**Step 06:** The oneM2M platform collects the received information, i.e., a list of available registrable oneM2M type devices and responds to the oneM2M application.

**Step 07:** The oneM2M application selects the device to be registered and requests registration of the selected oneM2M type device to the oneM2M platform.

**Step 08:** The oneM2M platform performs the registration request for the selected device.

**Step 09:** Assuming the oneM2M type device has an upper interface, the oneM2M platform sends a message to trigger registration to the selected device.

**Step 10:** The oneM2M type device receiving the registration triggering message performs the oneM2M resource registration procedure using the information (e.g., server platform address, Key, etc.) included in the message.

**Steps 11 ~ 13:** The oneM2M platform performs the received registration procedure and creates related resources.

### -----------------------End of change 1-------------------------------------------