Internet Engineering Task Force   Z. Cao
Internet-Draft   China Mobile
Intended status: Informational   A. Ding
Expires: February 28, 2014   Cambridge University / Helsinki University
August 27, 2013

Service Discovery in a Multiple Connection Environment: Problem Statement
draft-cao-mif-srv-dis-ps-03

Abstract

This document analyzes the problems of service discovery in a multiple connection environment. A multiple connection environment consists of multiple-interfaced nodes connecting to multiple networks or multiple provisioning domains. Given a type of service a multiple-interfaced client is looking for, the discovery progress ought to return a correct pointer to the service instance that the client is able to access without trying every available channel.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on February 28, 2014.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect

to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 2
2. Requirements and Terminology 3
2.1. Requirements 3
2.2. Terminology 3
3. Scenarios 3
3.1. Mif Scenario 3
3.2. Homenet Scenario 5
4. Problem Analysis 5
5. IANA Considerations 9
6. Security Considerations 9
7. References 9
7.1. Normative References 9
7.2. Informative References 9
Authors' Addresses 10

1. Introduction

A multihomed host has multiple provisioning domains via physical and/or virtual interfaces. A multihomed host receives node configuration information from each of its access networks, through various mechanisms such as DHCP, PPP and IPv6 Router Advertisements. When the received node-scoped configuration objects have different values from each administration domains, such as different DNS servers IP addresses, different default gateways or different address selection policies, the node has to decide which it will use or how it will merge them.

Issues regarding how the multi-homed host uses the configuration objects have been addresses in [RFC6418]. Current practices of how the various implementations handle these problems are introduced in [RFC6419]. [RFC6731] extends DHCPv6 to inform the host which DNS server it ought to select to send the query request, and DNS based Service Discovery (DNS-SD) has been specified in [RFC6763].

This document analyzes the problem of service discovery in a multiple connection environment. A multiple connection environment consists of multiple-interfaces nodes connecting to multiple networks or multiple provisioning domains. Given a type of service a multiple-interfaced client is looking for, the discovery progress ought to return a correct pointer to the service instance that the a client is able to access.

2. Requirements and Terminology

2.1. Requirements

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 [RFC2119].

2.2. Terminology

Service Domain

A set of services that can be accessed by users. Besides providing services, a service domain is responsible for delivering configuration and pointers that ensure a guaranteed service access.

Service Discovery

Procedure to acquire information that is necessary to access service.

Multiple Connection Environment

Consists of multiple-interfaced nodes that connect to multiple networks or multiple provisioning domains.

3. Scenarios

We describe two scenarios in this section, one related to Multiple Interfaces, and the other one related to Home Networks (homenet).

3.1. Mif Scenario

The service discovery process can be summarized as the following five steps.

1. Service Discovery Preparation: the host determines which interface it should send a query request based on the configuration information.

2. Service Query Request: the host sends a query request to find a service. The query should include a description of the service, for example, a full-qualified domain name, a URI, or an application-specific naming of the service.

3. Service Request Handling: any entity that receives the query request should handle the request. The entity should understand

the meaning of the request, and check the semantics of the request language before giving an answer back.

4. Service Query Response: the entity that receives the query request should reply with an answer to the query. The answer should include a pointer to the service.

5. Service Access: the host accesses the service via the pointer provided in the query response. The host is supposed to be able to get the service instance via the pointer under a successful and efficient service discovery mechanism, unless the servers in such service domain encounter problems e.g. a web server is down.

Figure 1 shows a typical scenario for service discovery in a multiple connection environment. It is common in today's mobile Internet that a host is equipped with multiple network interfaces. On the service domain, different services are deployed and some services may not be accessible to a certain interface on the host due to security concern or access policy. The connectivity each interface provides may not be restricted to Internet access. For instance, WLAN and bluetooth can offer direct access to potential services e.g. printers via local ad-hoc connectivity. In such multiple connection environment, the service discovery process should return a correct point to the host and ensure that the host can access the service via this pointer. This situation makes the multiple interface service discovery different from the typical one-interface Internet access scenario. Furthermore, the growing usage of IPv6 in the homenet environment has made service discovery more challenging that requires thorough investigation.

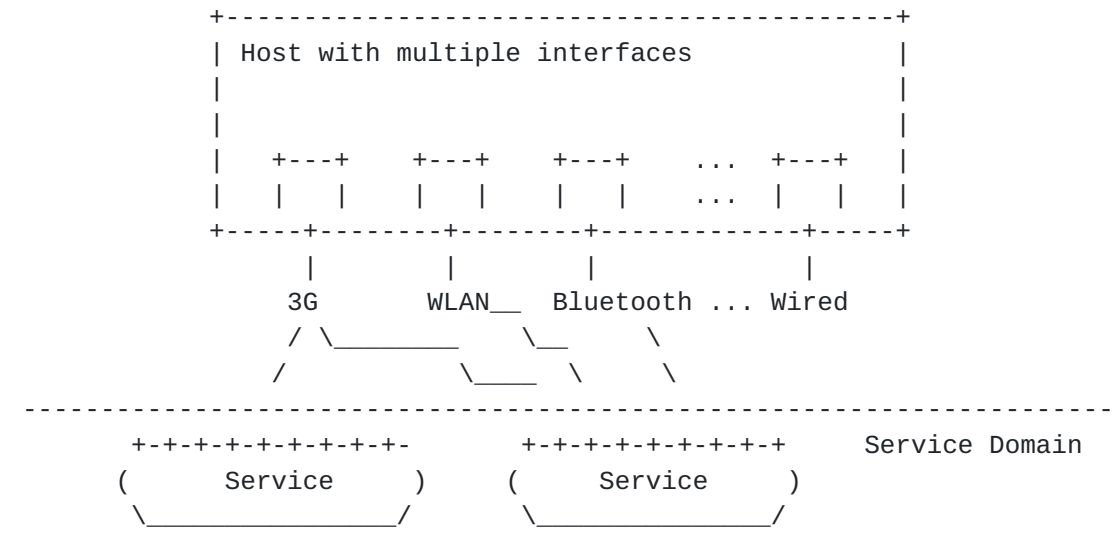


Figure 1: Multiple Interface Host with Multiple Available Services

3.2. Homenet Scenario

We also describe the issues related to the homenet architecture [I-D.ietf-homenet-arch], as depicted in Figure 2.

Suppose one MIF host is connected to three domains: homenet domain, 3gpp domain and a WiFi or enterprise domain. There is one service that is named with the private domain name, say 'temperature.ietf', which is only resolvable via the domain name service residing inside the homenet and is supported by the multicast dns service [RFC6762].

There are several problems in this scenario. First of all, since the host has two unicast dns domains configured over the 3GPP and WiFi, and as well as a multicast service discovery domain within the homenet, the host does not know which domain it should send a dns resolution request. Secondly, even if coupled with the split dns solution [RFC6731], the configuration information obtained from DHCP supports only those two unicast dns domains, but not the homenet domain which is normally considered as 'zero-configuration'. Third, the service discovery problem will become more complicated if the host is connecting to more than one home networks, i.e., multiple multicast dns domains.

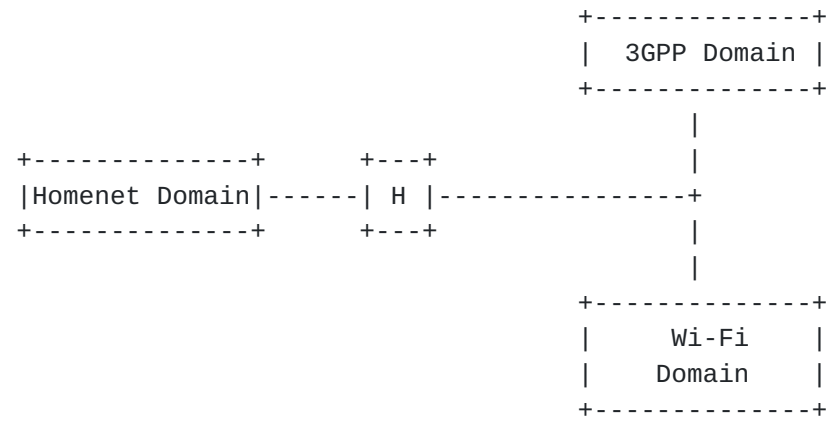


Figure 2: Homenet Scenario

4. Problem Analysis

The problems that a multiple-interfaced host may meet during the service discovery include:

1. How the query requests are sent? Because there are multiple interfaces available and multiple service rendezvous existing,

the host should decide which destination it ought to send the query to. And if there is a round-robin mechanism, the host should determine the order of the query request.

2. How to handle the query and reply? Some pointers to the service are restricted to a local scope or a certain interface, e.g., an office printer may not be accessible to the 3G-interface. The service discovery process is supposed to return a pointer that is accessible to the host.

3. How to access the service? Given the pointer to the service, the host should be able to determine from which interface it can access the service.

The existing work of [RFC6418] and [RFC6419] have covered the general problems encountered by hosts accessing multiple provisioning domains, but the focus is on connectivity and configuration. Proposal of Happy Eyeball in [I-D.ietf-mif-happy-eyeballs-extension] allows a host with multiple interfaces to pick a suitable one for access and enables automatic fallback. In a DNS based service discovery [RFC6763], the problem of domain split is analyzed in the [RFC6731]. The document defines an extension to the DHCPv4 and DHCPv6 to inform the MIF host which domain scope the Recursive DNS Server(RDNSS) is serving for, so that the "service query request" can be sent to the correct RDNSS to get an answer.

The existing proposals resolve the partial problem in the service discovery process mentioned above. To highlight the missing blocks, Figure 3 provides a 'gap' analysis. In the figure, we compare three existing solutions on service discovery, DNS-SD[RFC6763], DNS-Server-Selection [RFC6731], and MIF Happy Eyeball [I-D.ietf-mif-happy-eyeballs-extension], from three aspects as mentioned above. The DNS-Srv-Sel solution uses the defined DHCP option for the MIF host to select the corresponding DNS Server, and MIF-HE inherits this method in its most updated version. The MIF-HE can help host failover to the workable interface during service access while DNS-Srv-Sel does not handle this particular issue. The DNS-SD is not designed for a multiple interfaces environment and DNS server selection and request handling are based on standard DNS behaviors.

Aspects \Sol	DNS-SD	DNS-Srv-Sel	MIF-HE
How to Send Query	Std. DNS behavior	DHCP Option informed	Same as DNS-Srv-Sel

How to Handle Queries	Std. DNS server behavior	selection based on option	Same as DNS-Srv-Sel
How to Access service	no guarantee of connectivity	not possible if ports rejected	Failover to the Happiest one

Figure 3: Gap Analysis of Existing Service Discovery Methods

In a complicated network as shown in Figure 4 , the host connects to the enterprise network via the wired interface, a WLAN network with the 802.11 interface, and an operator's network via the cellular interface. The three intranets have their own Firewall policies to the global Internet. On the enterprise network, many outgoing ports are restricted, and on the WLAN and operator's public network, there is more freedom. If the MIF host makes a DNS-SRV query to a service in a global domain, all the RDNS servers have the corresponding records. But say the service port number has been blocked by the enterprise network administrator, the DNS has no such information. Even if the DNS returns a pointer to the MIF host, the MIF host cannot access this service via the wired interface.

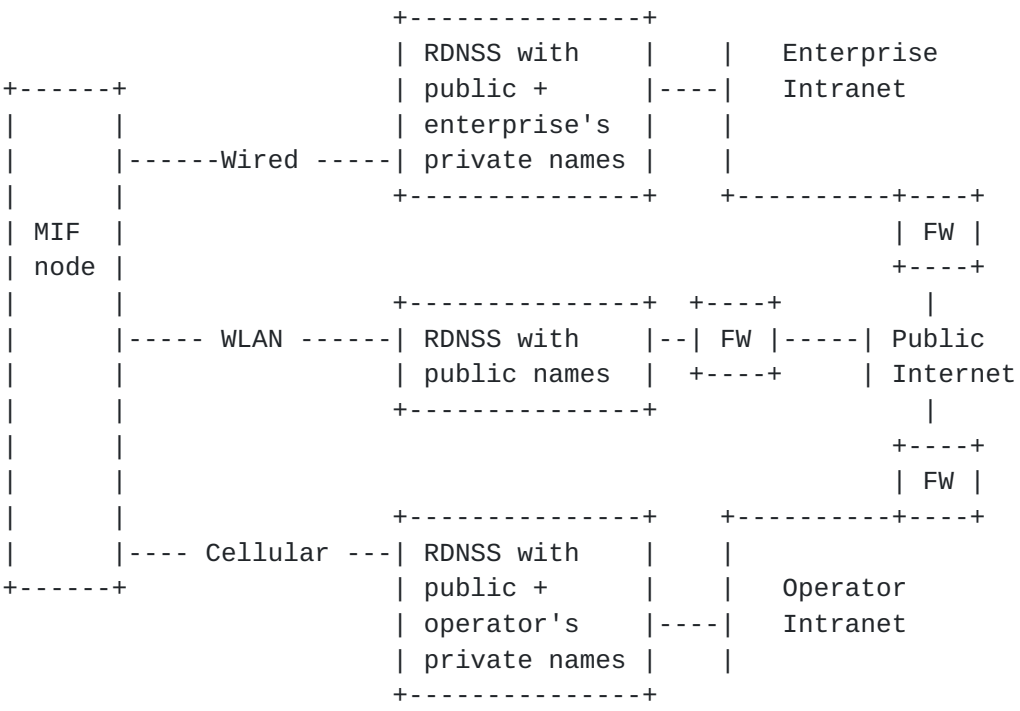


Figure 4: Scenario of Multiple Interface DNS Service Discovery

CoAP [I-D.ietf-core-coap] is an IETF designed RESTful protocol for constrianed environment. CoAP defines a link-format for service discovery of the particular CoAP server, i.e., "/.well-known/core". If the CoAP client has multiple access networks as shown in Figure 5, the situation turns to be more complex. For instance, if the MIF client wants to find a humidity sensing resource, but does not know which domain contains the information, it basically needs to send multiple CoAP GET requests with the well-known URL. Once it gets a response for the required resource, it can send the corresponding request to get the information. However this way is sub-optimal especially for constrained devices. MIF service discovery SHOULD consider the efficiency of the service discovery process.

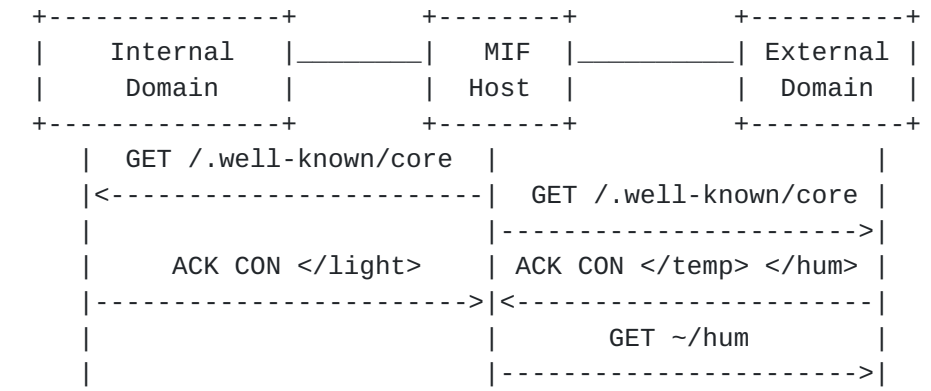


Figure 5: CoAP Service Discovery for a MIF Host

As a summary of the above analysis, the general problems and requirements of service discovery in a MIF environment can be summarized as follows:

Service Directory Service Configuration: Service directory is the entity that stores or can get the stored relationship between service names and service pointers. Different interfaces or provisioning domains have their different service directories. How to configure them on the MIF host and how the MIF host utilizes the configured information are important for the service discovery process to behave correctly.

Service Directory Selection: After the service directory information is configured on the host, the host is able to select the correct directory to send the query. The host can utilize auxiliary information available or send the query to all the directories that have been configured. The behavior of MIF host to select a correct directory is also important for a stable system.

Service Pointer/Address Resolution: The same service may have different available addresses and pointers, and some service has limited connectivity. So the resolution process should be able to return to the MIF host a record that is accessible from at least one of the interfaces. Efficiency SHOULD be taken into consideration in this phase.

Service Route Selection: With the pointers returned, the host should route the service level data to the service instance identified by the returned pointers.

5. IANA Considerations

This document has no IANA requests.

6. Security Considerations

The query response exchanges should be protected by security mechanisms. If the response contains invalid information, e.g. a pointer to a worm website, it harms. As a consequence, the service discovery should protect bogus information injected by attackers or intruders. The security consideration ought to be made by the underlining protocols, and it is out the scope of this problem statement document.

7. References

7.1. Normative References

[I-D.ietf-mif-happy-eyeballs-extension]
Chen, G., Williams, C., Wing, D., and A. Yourtchenko, "Happy Eyeballs Extension for Multiple Interfaces", draft-ietf-mif-happy-eyeballs-extension-02 (work in progress), February 2013.

[RFC6731] Savolainen, T., Kato, J., and T. Lemon, "Improved Recursive DNS Server Selection for Multi-Interfaced Nodes", RFC 6731, December 2012.

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

[RFC6763] Cheshire, S. and M. Krochmal, "DNS-Based Service Discovery", RFC 6763, February 2013.

7.2. Informative References

[I-D.ietf-core-coap]

Shelby, Z., Hartke, K., and C. Bormann, "Constrained Application Protocol (CoAP)", draft-ietf-core-coap-18 (work in progress), June 2013.

[I-D.ietf-homenet-arch]
Chown, T., Arkko, J., Brandt, A., Troan, O., and J. Weil, "Home Networking Architecture for IPv6", draft-ietf-homenet-arch-10 (work in progress), August 2013.

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

[RFC6418] Blanchet, M. and P. Seite, "Multiple Interfaces and Provisioning Domains Problem Statement", RFC 6418, November 2011.

[RFC6419] Wasserman, M. and P. Seite, "Current Practices for Multiple-Interface Hosts", RFC 6419, November 2011.

Authors' Addresses

Zhen Cao
China Mobile
Xuanwumenxi Ave. No. 32
Beijing 100871
China

Phone: +86-10-52686688
Email: zehn.cao@gmail.com, caozhen@chinamobile.com

Aaron Yi Ding
Cambridge University / Helsinki University
William Gates Building, 15 JJ Thomson Ave
CB3 0FD Cambridge
United Kingdom

Phone: +44-7934034801; +358-9-19151296
Email: Aaron.Ding@cl.cam.ac.uk