Well known site local unicast addresses for DNS resolver
<draft-ietf-ipv6-dns-discovery-05.txt>

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC2026.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet Drafts are 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 a "work in progress".

To view the list Internet-Draft Shadow Directories, see http://www.ietf.org/shadow.html.

Abstract

This document specifies a method for nodes to find a DNS resolver with minimum configuration in the network and without running a discovery protocol on the nodes.

Copyright Notice

Copyright (C) The Internet Society (2002). All Rights Reserved.

1. Introduction

RFC 2462 [ADDRCONF] provides a way to autoconfigure nodes with one or more IPv6 address and default routes.

However, for a node to be fully operational on a network, many other parameters are needed, such as the address of a DNS resolver, mail relays, web proxies, etc. Except for name resolution, all the other services are usually described using names, not addresses, such as smtp.myisp.net or webcache.myisp.net. For obvious bootstrapping reasons, a node needs to be configured with the IP address (and not the name) of a DNS resolver. As IPv6 addresses look much more complex than IPv4 ones, there is some incentive to make this
configuration as automatic and simple as possible. Although it would be desirable to have all configuration parameters configured/discovered automatically, it is common practice in IPv4 today to ask the user to do manual configuration for some of them by entering server names in a configuration form. So, a solution that will allow for automatic configuration of the DNS resolver is seen as an important step forward in the autoconfiguration story.

The intended usage scenario for this proposal is a home or enterprise network where IPv6 nodes are plugged/unplugged with minimum management and use local resources available on the network to autoconfigure. This proposal is also useful in cellular networks where all mobile devices are included within the same site.

2. Pre-configuration vs discovery

Some of the discussions in the past around DNS server discovery have been trying to characterize the solution space into stateless versus stateful or server-oriented versus serverless. It is not absolutely clear how much state if any needs to be kept to perform DNS server discovery, and, although the semantic differences between a router and a server are well understood from a conceptual perspective, the current implementations tend to blur the picture. In another attempt to characterize different approaches, one can look at how much intelligence a client needs to have in order to use the service.

One avenue is to ask the IPv6 node to participate in a discovery protocol, such as SLP or DHCP, learn the address of the server and send packets to this server. Another one is to pre-configure (hard-code) a local scope address on the IPv6 node and let it send packets directly to this address, with the underlying assumption that the routing system will forward them to the right place. This document explores this later avenue of pre-configuration that does not require participation of the end node in the DNS resolver discovery mechanism.

The mechanism described here is to be used as a last resort, when no other configuration information is available.

3. Reserved Prefix and addresses

The basic idea of this proposal is to reserve a well known IPv6 site local prefix and three well known IPv6 addresses for DNS resolvers and then have the routing system forward the DNS request to those DNS resolvers.

IPv6 nodes will be pre-configured (hard coded) with those three IPv6 addresses as DNS resolver.

Each local DNS resolvers should be configured with one of those three addresses to enable clients to switch from one to the other if one fails. Host routes for each of those resolvers should be injected in the local routing system. Example methods for injecting host routes and a brief discussion of their fate sharing properties is presented.
a) Manual injection of routes by a router on the same subnet. If the node running the DNS resolver goes down, the router may or may not be notified and keep announcing the route.

b) Running a routing protocol on the same node running the DNS resolver. If the process running the DNS resolver dies, the routing protocol may or may not be notified and keep announcing the route.

c) Running a routing protocol within the same process running the DNS resolver. If the DNS resolver and the routing protocol run in separated threads, similar concerns as above are true.

d) Having an "announcement" protocol that the DNS resolver could use to advertise the host route to the nearby router. Details of such a protocol are out of scope of this document, but something similar to [MLD] is possible.

IANA considerations for this prefix are covered in Section 6.

4. Site local versus global scope considerations

The rationales for having a site local prefix are:

- a) Using a site local prefix will ensure that the traffic to the DNS resolver stays local to the site. This will prevent the DNS requests from accidentally leaking out of the site. However, the local resolver can implement a policy to forward DNS resolution of non-local addresses to an external DNS resolver.

- b) Reverse DNS resolution of site local addresses is only meaningful within the site. Thus, making sure that such queries are first sent to a DNS resolver located within the site perimeter increase their likelihood of success.

Note: there is currently some discussions about the usefulness of site local addresses in the IPv6 architecture. Depending on the outcome of this discussion, this section will need to be revisited. If a global prefix was chosen for this mechanism, concerns raised in a) could be addressed using a simple access list on the site exit routers and concerns raised in b) would disappear.

5. Examples of use

This section presents example scenarios showing how the mechanism described in this memo can co-exist with other techniques, namely manual configuration and DHCPv6 discovery.

5.1 Simple case, general purpose DNS resolver

This example shows the case of an enterprise or a cellular network
that manages a full flavor general purpose DNS resolver and a large number of nodes running DNS stub resolvers. The DNS resolver is performing (and caching) all the recursive queries on behalf of the stub resolvers. Those stub resolvers are either manually configured with the IPv6 address of the resolver or with one (or several) of the well known site local unicast addresses defined in this memo.

(The DNS resolver is configured to listen both on its IPv6 address and on the well known address)

5.2 DNS forwarder

A drawback of the choice of site local scope for the reserved addresses for DNS resolver is that, in the case of a home/small office network connected to an ISP, DNS traffic cannot be sent directly to the ISP DNS resolver without having the ISP and all its customers share the same definition of site.

In this scenario, the home/small office network is connected to the ISP router (PE) via an edge router (CPE). Prefix delegation is performed out of band is is out of scope of this memo.

The customer router CPE could be configured on its internal interface with one of the reserved site local addresses and listen for DNS queries. It would be configured to use one (or several) of the well known site local unicast addresses within the ISP’s site to send its own queries to. It would act as a DNS forwarder, forwarding queries received on its internal interface to the ISP’s DNS resolver.
In this configuration, the CPE is acting as a multi-sited router.

5.3 DNS forwarder with DHCPv6 interactions

In this variant scenario, DHCPv6 could be used between the PE and CPE to do prefix delegation [DELEG] and DNS resolver discovery.

This example will show how DHCPv6 and well known site local unicast addresses can be used at the same time within a site to discover the address of the DNS forwarder.

The customer router CPE could be configured on its internal interface with one of the reserved site local addresses and listen for DNS queries. It would act as a DNS forwarder, forwarding those queries to the DNS resolver pointed out by the ISP in the DHCPv6 exchange.

The same CPE router could also act as a local DHCPv6 server, advertising either itself as DNS forwarder.
Within the site:

a) DHCPv6 aware clients could use DHCPv6 to obtain the address of the DNS forwarder...

b) other nodes may simply send their DNS request to the reserved site local addresses.

A variant of this scenario is the CPE can decide to pass the global address of the ISP DNS resolver in the DHCPv6 exchange with the internal nodes.

6. IANA considerations

The site local prefix fec0:0000:0000:ffff::/64 is to be reserved out of the site local fec0::/10 prefix.

The unicast addresses fec0:000:0000:ffff::1, fec0:000:0000:ffff::2 and fec0:000:0000:ffff::2 are to be reserved for DNS resolver configuration.

All other addresses within the fec0:0000:0000:ffff::/64 are reserved for future use and are expected to be assigned only with IESG approval.

7. Security Considerations

Ensuring that queries reach a legitimate DNS server relies on the security of the IPv6 routing infrastructure. The issues here are the same as those for protecting basic IPv6 connectivity.

IPsec/IKE can be used as the well-known addresses are used as unicast
addresses.

The payload can be protected using standard DNS security techniques. If the client can preconfigure a well known private or public key then TSIG [TSIG] can be used with the same packets presented for the query. If this is not the case, then TSIG keys will have to be negotiated using [TKEY]. After the client has the proper key then the query can be performed.

The use of site local addresses instead of global addresses will ensure the DNS queries issued by host using this mechanism will not leak out of the site.

8. References

[ADDRCONF]

[MLD]

[TSIG]

[TKEY]

[DHCPv6]

[DELEG]

9. Authors’ Addresses

Alain Durand
SUN microsystems, inc.
901 San Antonio rd UMPK 17-202
Palo Alto, CA 94303, USA.
Email: Alain.Durand@sun.com

Jun-ichiro iotjun HAGINO
Research Laboratory, Internet Initiative Japan Inc.
Takebashi Yasuda Bldg.,
10. Full Copyright Statement

Copyright (C) The Internet Society (2002). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.