

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: April 22, 2010

M. Boucadair
E. Burgey
France Telecom
October 19, 2009

DNS64 Service Location and Discovery
draft-boucadair-behave-dns64-discovery-00

Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of [BCP 78](#) and [BCP 79](#). This document may contain material from IETF Documents or IETF Contributions published or made publicly available before November 10, 2008. The person(s) controlling the copyright in some of this material may not have granted the IETF Trust the right to allow modifications of such material outside the IETF Standards Process. Without obtaining an adequate license from the person(s) controlling the copyright in such materials, this document may not be modified outside the IETF Standards Process, and derivative works of it may not be created outside the IETF Standards Process, except to format it for publication as an RFC or to translate it into languages other than English.

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 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."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/lid-abstracts.txt>.

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

This Internet-Draft will expire on April 22, 2010.

Copyright Notice

Copyright (c) 2009 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

Internet-Draft

DNS64 Service Discovery

October 2009

Provisions Relating to IETF Documents in effect on the date of publication of this document (<http://trustee.ietf.org/license-info>). Please review these documents carefully, as they describe your rights and restrictions with respect to this document.

Abstract

This memo proposes a set of solutions for the discovery of DNS64 [[I-D.ietf-behave-dns64](#)] service. Three solution candidates are proposed: SRV RR, DHCP and RA-based.

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](#)].

Table of Contents

1.	Introduction	4
1.1.	Overall Context	4
1.2.	Technical Issues	4
1.3.	Contribution of this Draft	5
2.	Terminology	5
3.	Discovery of DNS64 Service	6
3.1.	DNS-based Approach: DNS64 SRV RR	6
3.1.1.	Rationale	6
3.1.2.	DNS64 SRV RR Definition	7
3.1.3.	Example Usage	8
3.2.	DHCP-based Approach	8
3.3.	RA-based Approach	9
4.	On the Location of A64_DNS and AAAA_DNS Functions	9
5.	IANA Considerations	10
6.	Security Considerations	10
7.	Acknowledgements	10
8.	References	10
8.1.	Normative References	10
8.2.	Informative References	11
	Authors' Addresses	12

[1.](#) Introduction

[1.1.](#) Overall Context

Recently, due to IPv4 address depletion, several solutions have been proposed within IETF. Among those solutions, IPv6 is the only perennial solution that should be implemented by service providers. Nevertheless, this implementation won't be in one shot and the co-existence between IPv4 and IPv6 should be managed for a while. In addition to the co-existence issue, interconnection means to enable successful communications between IPv4 and IPv6 realms must be enforced.

In this context, BEHAVE WG is currently specifying translator mechanisms which cover both stateful [\[I-D.ietf-behave-v6v4-xlate-stateful\]](#) and stateless [\[I-D.ietf-behave-v6v4-xlate\]](#) modes. The proposed solutions require the definition of a new IPv4-mapped IPv6 address [\[I-D.ietf-behave-address-format\]](#) which is used to represent an IPv4-only host in an IPv6 realms and the definition of a new mechanism called DNS64 for synthesizing a AAAA record, representing an IPv4-mapped IPv6 address in the DNS system, from the A record representing the IPv4 address of the IPv4-only host [\[I-D.ietf-behave-dns64\]](#).

This IPv4-mapped IPv6 address, when managed by a translator, is to be considered as "fake" address in a DNS system since it is not necessarily assigned to any host's interface qualified by the associated name. In fact, the IPv4-mapped IPv6 address represents the IPv6 interface of the translator used to translate packet

exchanged with the IPv4 host.

Moreover, it can be confusing for applications since the application querying for this address must be aware that the application running on the host represented by the IPv4-mapped IPv6 address is connected through an IPv4 interface and may be not able to use a reference [[I-D.carpenter-behave-referral-object](#)] to an IPv6 address in the packet payload.

When an IPv4-mapped IPv6 address is used as destination address, the traffic will be handled by a translator device and no direct communication would be possible.

[1.2.](#) Technical Issues

This memo discusses issues that may be encountered when deploying DNS64 mechanism in operational networks. Particularly, In order to avoid crossing NAT64 boxes, natives communications should be encouraged. Therefore, an IPv6 host should be able to prefer a

native destination IPv6 address to reach a remote peer instead of any IPv4-mapped IPv6 address synthesised through the DNS64 mechanism.

Furthermore, a dual-stack host should be able to prefer the native IPv4 address instead of any IPv4-mapped IPv6 address but should be able to prefer the native IPv6 address instead of any IPv4 one. Consequently, the activation of the DNS64 mechanism must be avoided for dual-stack hosts if they are not able to distinguish native AAAA records from synthesised AAAA records.

A solution to distinguish native AAAA records from synthesised AAAA records is proposed in [[I-D.boucadair-behave-dns-a64](#)]. This proposal introduces a new A64 DNS Record Type for synthesised AAAA records. But, as this new record type will not be "understood" by existing implementations, it is necessary, during a transition phase, to return, to non-updated IPv6 applications or non updated IPv6 DNS stub resolvers, AAAA records for synthesised AAAA records.

Therefore the original DNS64 mechanism is updated as follows:

- o Rely on DNS64 mechanism as described in [[I-D.ietf-behave-dns64](#)]. When supported, a DNS server returns AAAA records for native IPv6

addresses records and IPv4-mapped IPv6 addresses. This mechanism may be used during the transition phase.

- o Introduce A64-capable DNS which returns AAAA records for native IPv6 addresses and A64 records for IPv4-mapped IPv6 addresses.

[1.3.](#) Contribution of this Draft

This document focuses on the following issues:

- o Discovery of a compliant DNS server which is able to generate synthesised AAAA records [[I-D.ietf-behave-dns64](#)].
- o Ability to distinguish native AAAA records from synthesised AAAA records.

[2.](#) Terminology

This document makes use of the following terms.

- o Authoritative server: A DNS server that can answer authoritatively for a given DNS query.
- o Stub resolver: A resolver with minimum functionality, typically for use in end points that depends on a recursive resolver. Most

end points on the Internet use stub resolvers.

- o Recursive resolver: A DNS server that accepts requests from one resolver, and asks another resolver for the answer on behalf of the first resolver.
- o AAAA_DNS function: A logical function that synthesizes AAAA records (containing IPv6 addresses) from A64 records if there is no AAAA native records and generate also AAAA records from A records(containing IPv4 addresses) if there is no AAAA native records and no A64 records for the requested domain name.
- o A64_DNS function: A logical function that returns A64 records or synthesizes A64 records (containing IPv6 addresses) from A records (containing IPv4 addresses).

- o AAAA_DNS server: A DNS server which supports AAAA_DNS function.
- o A64_DNS server: A DNS server which supports A64_DNS function.
- o A64_DNS recursor: A recursive resolver that provides the A64_DNS function as part of its operation.
- o Synthetic RR: A DNS resource record (RR) that is not contained in any zone data file, but has been synthesized from other RRs. An example is a synthetic AAAA record created from an A record.
- o DNS64 service: Denotes the service offered by a DNS server which is supporting AAAA_DNS or A64_DNS functions.
- o A DNS server that includes neither AAAA_DNS nor A64_DNS function and is not able to manage A64 records is called a "DNS64 uncompliant DNS server".

[3.](#) Discovery of DNS64 Service

[3.1.](#) DNS-based Approach: DNS64 SRV RR

[3.1.1.](#) Rationale

This section proposes to define a new SRV RR [[RFC2782](#)] for the location of DNS64 service. DNS64 service is unambiguously distinguished from native AAAA-capable DNS servers. DNS64 service should not be invoked for the resolution of native AAAA records. A decision-making process may be enforced in the client side to drive the invocation of DNS64 service or native IPv6 DNS server appropriately. Otherwise, the traffic which may go through the

translators is not optimised and extra-processing would be required.

The DNS64 service can also be defined using S-NAPTR [[RFC3958](#)] or U-NAPTR [[RFC4848](#)].

[3.1.2.](#) DNS64 SRV RR Definition

The format of the generic SRV RR defined in [[RFC2782](#)] is as follows:

`_Service._Proto.Name TTL Class SRV Priority Weight Port Target`

Within this memo, only the description of "Service" and "Target" are re-produced below. For more information about the description of the fields, the reader is invited to refer to [[RFC2782](#)].

- o Service: The symbolic name of the desired service. For the DNS64 service, "DNS64" is proposed as a service name for the location of DNS64 service. If the proposal is adopted, a formal service name is to be assigned by IANA. When used in an SRV RR, the assigned service name must be prepended with an "_" character.
- o Target: The domain name of the target host. There must be one or more address records for this name. A Target of "." means that the service is decidedly not available at this domain.

In order to retrieve a the IP address of a DNS server supporting the DNS64 service, the client should be SRV-capable and should be configured to issue a dedicated SRV query. The result of the corresponding query should be used to retrieve the synthesised IPv6 addresses representing IPv4 hosts.

A host should be provided with DNS servers which support AAAA records and servers which support DNS64 service. In order to avoid invoking NAT64 boxes, a client should be configured in order to prefer requesting native IPv6 DNS servers first. If not result has been retrieved, DNS64 can be then invoked.

This sequential approach may increase the required delay for the establishment of a communication. Several solutions (out o scope of this document) can be envisaged such as:

1. Issue simultaneous queries;
2. Define a new record type/query type to identify an IPv4-mapped IPv6 address from a native IPv6 one;
3. Define a new query type to retrieve all existing records.

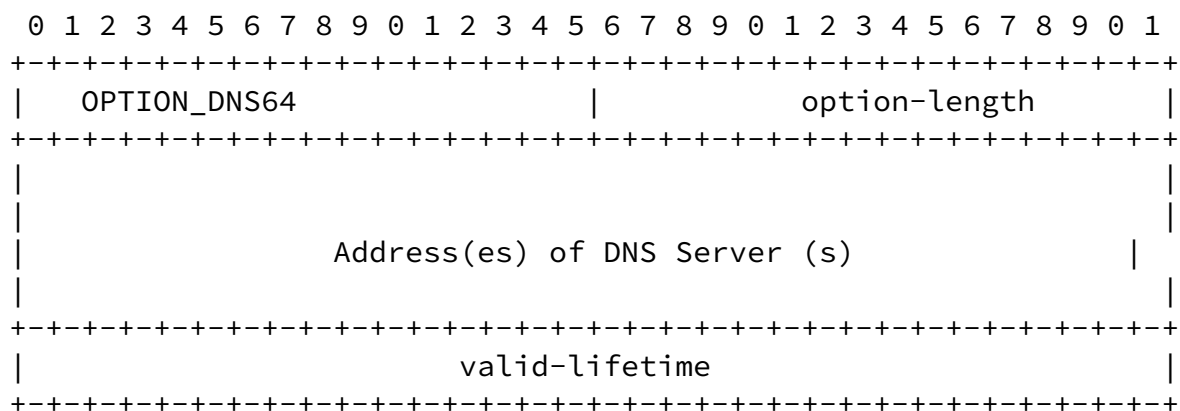
A client is provisioned with the QNAME to be used for the resolution of DNS64 service.

If a client wants to discover a DNS server that provides DNS64 service for the domain MyInternetAccessDomain.com., it does a lookup of _DNS64._udp.MyInternetAccessDomain.com.

3.2. DHCP-based Approach

DHCPv6 can be used to provision the IP address(es) of DNS server(s) which are capable to provide DNS64 service. A new option is required for this purpose.

The format of the proposed DHCPv6 Option is shown in the figure hereafter:



Below is provided the description of the fields:

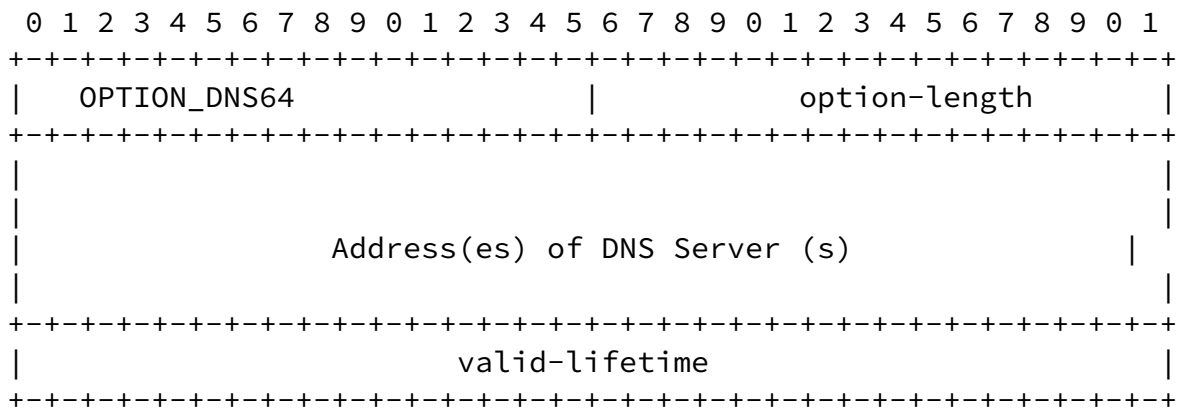
- o Type: OPTION_DNS64. To be assigned by IANA for DNS64 service.
- o option-length: variable
- o Address(es) of DNS Server (s): One or a list of IP addresses of DNS server(s) supporting the DNS64 service.
- o valid-lifetime: The valid lifetime for the address(es) enclosed in the option, expressed in units of seconds.

TBC.

3.3. RA-based Approach

This section describes a RA-based solution which is similar to the one described in [[RFC5006](#)] for the discovery of RRDNS servers.

This option is used to convey the IPv6 address of the DNS server (s) supporting the DNS64 service.



Below is provided the description of the fields:

- o Type: OPTION_DNS64. To be assigned by IANA for DNS64 service.
- o option-length: variable
- o Address(es) of DNS Server (s): One or a list of IP addresses of DNS server(s) supporting the DNS64 service.
- o valid-lifetime: The valid lifetime for the address(es) enclosed in the option, expressed in units of seconds.

TBC.

4. On the Location of A64_DNS and AAAA_DNS Functions

AAAA_DNS function should be located in a stub resolver or may be located in the first recursive resolver after the stub resolver.

AAAA_DNS function should not be activated in a stub resolver located on a dual-stack host with access to public IPv4 and IPv6 networks. An exception to this rule may happen to manage communication between IPv6-only application on dual-stack host and IPv4-only application, but such case may not be common.

AAAA_DNS function should not be activated in a resolver called by a

stub resolver located on a dual-stack host with access to public IPv4 and IPv6 networks. AAAA_DNS function must not be activated on an authoritative server.

A64 records, may be used on authoritative DNS servers, but only if the IPv4-mapped IPv6 addresses stored in this record has a global scope. In that case this records may be introduced in the server with standard provisioning tools, but in some cases it may be preferable to provision only the A records and to use an A64_DNS function to generate the corresponding A64 records.

More generally, an A64_DNS function should be located in a recursive DNS resolver chosen so that, for all IPv4-mapped IPv6 addresses stored in A64 records generated by this A64_DNS function, the translator interface represented by this IPv6 address is the best suitable to translate packets exchanged between any of the hosts using a stub resolver sending requests to this DNS resolver and the IPv4 interface represented by the IPv4 address mapped in this IPv6 address.

[5.](#) IANA Considerations

This memo requests the use of "_DNS64" service label. This service is associated only with UDP transport.

[6.](#) Security Considerations

Security considerations discussed in [[RFC2782](#)] should be taken into account.

[7.](#) Acknowledgements

TBC

[8.](#) References

8.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC2782] Gulbrandsen, A., Vixie, P., and L. Esibov, "A DNS RR for specifying the location of services (DNS SRV)", [RFC 2782](#), February 2000.

Boucadair & Burgey

Expires April 22, 2010

[Page 10]

Internet-Draft

DNS64 Service Discovery

October 2009

- [RFC3646] Droms, R., "DNS Configuration options for Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", [RFC 3646](#), December 2003.
- [RFC3958] Daigle, L. and A. Newton, "Domain-Based Application Service Location Using SRV RRs and the Dynamic Delegation Discovery Service (DDDS)", [RFC 3958](#), January 2005.
- [RFC4339] Jeong, J., "IPv6 Host Configuration of DNS Server Information Approaches", [RFC 4339](#), February 2006.
- [RFC4848] Daigle, L., "Domain-Based Application Service Location Using URIs and the Dynamic Delegation Discovery Service (DDDS)", [RFC 4848](#), April 2007.
- [RFC5006] Jeong, J., Park, S., Beloeil, L., and S. Madanapalli, "IPv6 Router Advertisement Option for DNS Configuration", [RFC 5006](#), September 2007.

8.2. Informative References

- [I-D.carpenter-behave-referral-object]
Carpenter, B., Boucadair, M., Brim, S., Halpern, J., Jiang, S., and K. Moore, "A Generic Referral Object for Internet Entities", [draft-carpenter-behave-referral-object-00](#) (work in progress), May 2009.
- [I-D.ietf-behave-address-format]
Huitema, C., Bao, C., Bagnulo, M., Boucadair, M., and X. Li, "IPv6 Addressing of IPv4/IPv6 Translators", [draft-ietf-behave-address-format-00](#) (work in progress),

August 2009.

[I-D.ietf-behave-dns64]

Bagnulo, M., Sullivan, A., Matthews, P., and I. Beijnum, "DNS64: DNS extensions for Network Address Translation from IPv6 Clients to IPv4 Servers", [draft-ietf-behave-dns64-00](#) (work in progress), July 2009.

[I-D.ietf-behave-v6v4-xlate]

Li, X., Bao, C., and F. Baker, "IP/ICMP Translation Algorithm", [draft-ietf-behave-v6v4-xlate-01](#) (work in progress), September 2009.

[I-D.ietf-behave-v6v4-xlate-stateful]

Bagnulo, M., Matthews, P., and I. Beijnum, "NAT64: Network Address and Protocol Translation from IPv6 Clients to IPv4

Boucadair & Burgey

Expires April 22, 2010

[Page 11]

Internet-Draft

DNS64 Service Discovery

October 2009

Servers", [draft-ietf-behave-v6v4-xlate-stateful-02](#) (work in progress), October 2009.

Authors' Addresses

Mohamed Boucadair
France Telecom
3, Av Francois Chateau
Rennes, 35000
France

Email: mohamed.boucadair@orange-ftgroup.com

Eric Burgey
France Telecom
Paris,
France

Email: eric.burgey@orange-ftgroup.com

