Abstract

This document describes an idea for a special DNS query whose use is to get the IPv6 address representing an IPv4 address in a 464xlat environment.

The query can also be used to force the IPv4 client to connect to the server via IPv6 by returning the "real" IPv6 address of a dual-stack server instead of the IPv6 address used to connect to the server's IPv4 address using NAT64.

The query is supposed to be compatible to the existing DNS system so no changes to the DNS protocol or DNS servers need to be done.

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 https://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 March 27, 2019.

Copyright Notice

Copyright (c) 2018 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 (https://trustee.ietf.org/license-info) in effect on the date of
1. Introduction

Because of the IPv4 address shortage the IPv6 protocol has been developed. Unfortunately many servers in the internet are still IPv4-only and many internet service providers are not able to assign an IPv4 address to every customer.

There is also software which is not able to use IPv6.

Many internet service providers use NAT64 [RFC6146] to provide their customers the possibility to use IPv4-only software to access the internet or to access IPv4-only servers using an IPv6-only network.

The IPv6 prefix 64:ff9b::/96 is reserved for calculating IPv6 addresses representing IPv4 addresses.

However there are advantages when not using this addressing scheme but when calculating the IPv6 address representing the IPv4 address on the internet service provider side:

First the internet service provider may use multiple NAT64 routers and do a load balancing by assigning different /96 prefixes to each NAT64 router and returning an IPv6 address based on the router with the least load to the customer.

Second the provider may return the real IPv6 address of a dual-stack server if that address is known. Doing so the load of the NAT64 routers can be reduced.

A third use case is to use different NAT64 routers based on different IPv4 addresses: To connect to an IPv4 server in the USA a NAT64 router in the USA may be used while a NAT64 in Europe is used to connect to a server in Europe.

There were already internet-drafts in the past which were addressing this problem.

This document describes a method based on DNS queries allowing the IPv6 address to be calculated by the internet service provider. The method also allows server operators of dual-stack servers to inform...
internet service providers about the IPv6 address of the server based on the IPv4 address.

2. Terminology

2.1. Keywords in capital letters

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14, RFC 2119 [RFC2119] and indicate requirement levels for compliant implementations.

2.2. client, server

In the context of this document a "client" is a node which is initiating data transfer between itself and another node. The other node is called "server" in the context of this document.

2.3. 464xlat

In the context of this document "464xlat" is a method of data transmission working the following way:

An internet service provider provides IPv6 only and it provides access to IPv4 servers via NAT64. However some application software or client hardware does not support IPv4 or for a certain server only the IPv4 address (but not the host name) is known.

The hardware and software on the customer side (such as a home router) is able to use an IPv6 connection to a NAT64 router to establish a connection between the IPv4-only software or hardware at the customer side and the IPv4 server in the internet.

Note that this definition is much more generic than what is typically understood by the term "464xlat".

3. Basic DNS query

This document suggests to use an AAAA query for the host name "<IPv4>.in-addr.arpa" to get the IPv6 address representing a certain IPv4 address.

To establish a connection to the server 192.0.2.34 the client sends a DNS request with the query "AAAA 34.2.0.192.in-addr.arpa" to the DNS server of the internet service provider.
The DNS server will NOT process the response normally but it will detect that the host name ends with ".in-addr.arpa" and respond with the IPv6 address that is used to connect to 192.0.2.34 using a NAT64 router. This may be 64:ff9b::192.0.2.34.

4. Use of DNS queries for dual-stack servers

Operators of dual-stack servers might add an AAAA record to their name server. The following example shows such an AAAA record:

34.2.0.192.in-addr.arpa. 3600 IN AAAA 2001:db8::234

Figure 1: Special AAAA record

The DNS servers of the internet service providers may now ask for the IPv6 address of a server by its IPv4 address the following way:

<table>
<thead>
<tr>
<th>Client</th>
<th>Provider’s DNS server</th>
<th>Server operator’s DNS server</th>
</tr>
</thead>
<tbody>
<tr>
<td>192.0.2.34.in-addr.arpa</td>
<td></td>
<td></td>
</tr>
<tr>
<td>64:ff9b::c000:222</td>
<td></td>
<td></td>
</tr>
<tr>
<td>192.0.2.34.in-addr.arpa</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2001:db8::234</td>
<td></td>
<td></td>
</tr>
<tr>
<td>192.0.2.34.in-addr.arpa</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2001:db8::234</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 2: Time line with special AAAA records

The client wants to establish a connection to 192.0.2.34. For this reason it sends a DNS query (1) to the internet service provider’s DNS server.

The internet service provider’s DNS server does not know the specific IPv4 address, yet. Therefore it answers with the IPv6 address (2) which is used to connect to the IPv4 server via NAT64.
The same time it asks the server operator’s DNS server for the special AAAA entry (3).

If the server is a dual-stack server and the server operator supports special AAAA records the DNS server will return the IPv6 address (4) of the dual-stack server.

Next time a client asks for the IPv6 address for connecting to an IPv4 host via NAT64 (5) the internet service provider’s DNS server does not return a IPv6 address using a NAT64 server but it directly returns the IPv6 address of the server.

This typically makes the connection faster and reduces the load of the NAT64 routers.

5. Special IPv6 addresses

The address range 64:ff9b::f000:0/100 lies into the address range reserved for NAT64 however these addresses are not valid because the addresses in the range 240.0.0.0/4 are not valid.

This can be used to return special status messages using AAAA records.

A client MUST assume that an address in this range means: "No connection possible" unless it is able to understand the special meaning of the address.

The provider’s DNS server MUST assume that an address in this range means: "Connection only possible via IPv4" unless it is able to understand the special meaning of the address.

For special IPv6 addresses in answers this document suggests the following form:

```
+---+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+---/+-+-+-
| 64:ff9b::f000:0/100 | Code | A | More options |
+---/+---+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+---/+-+-+-
```

Figure 3: Special addresses for AAAA records

- The first 100 bits of the IPv6 address are the constant prefix 64:ff9b::f000:0/100.

- The next 8 bits are the "status code". If a client receives an unknown status code it MUST assume that a connection via NAT64 is not possible. If a special AAAA record contains an unknown status code the provider’s DNS MUST assume that a connection is only
possible via IPv4. The provider’s DNS must only send status codes to the client if it is known that a certain status code is understood by the client.

- The next bit following is the "all" bit. If this bit is set all functionality accessible using this IPv4 address can also be accessed using "special AAAA records".

- The last 19 bits are reserved for future use. They MUST be set to zero and they MUST be ignored until their meaning is defined.

This document suggests the following "status codes":

- 1: This code is only sent by the provider’s DNS server and it is not found in "special AAAA records". The provider’s DNS server responds with the IP address 64:ff9b::f010:0 to indicate that different NAT64 routers for different higher-layer protocols (UDP, TCP) are used. The client shall add the higher-layer protocol to the pseudo host name to get the IPv6 address. Example: "tcp.34.2.0.192.in-addr.arpa" will return the IPv6 address for connecting to 192.0.2.34 using TCP.

- 2: This code works similar to code 1. However it indicates that different IPv6 addresses for TCP and UDP will be returned for this IPv4 address only. Unlike code 1 this code is valid in "special AAAA records".

- 3: This code indicates that different IPv6 addresses will be returned for different port numbers (such as TCP ports). This can be the case if different servers are behind a NAT and "port forwarding" is used. The port number and the protocol shall be used to query for the IPv6 address. Example: "80.tcp.34.2.0.192.in-addr.arpa" is used to get the IPv6 address for connecting to TCP port 80 of 192.0.2.34.

6. Example special AAAA records

6.1. Example 1: The simple case

34.2.0.192.in-addr.arpa. 3600 IN AAAA 2001:db8::234

Figure 4: AAAA records for example 1

The server has the IP addresses 2001:db8::234 and 192.0.2.34.

All TCP and UDP ports which are reachable via IPv4 can also be reached via IPv6.
(Note that there may be services - e.g. TCP ports - which are only reachable via IPv6.)

6.2. Example 2: Protocol specific

```
34.2.0.192.in-addr.arpa. 3600 IN AAAA 64::ff9b::f020:0
tcp.34.2.0.192.in-addr.arpa. 3600 IN AAAA 2001:db8::234
```

Figure 5: AAAA records for example 2

The server has the IP addresses 2001:db8::234 and 192.0.2.34.

All TCP ports which are reachable via IPv4 can also be reached via IPv6.

However the UDP ports which are reachable via IPv4 cannot be reached via IPv6.

6.3. Example 3: Port specific

```
34.2.0.192.in-addr.arpa. 3600 IN AAAA 64::ff9b::f030:0
23.tcp.34.2.0.192.in-addr.arpa. 3600 IN AAAA 2001:db8::234
80.tcp.34.2.0.192.in-addr.arpa. 3600 IN AAAA 2001:db8::5678
```

Figure 6: AAAA records for example 3

The two servers with the addresses 2001:db8::234 and 2001:db8::5678 are probably behind a NAT using port forwarding which has the address 192.0.2.34.

A connection to TCP port 23 of 192.0.2.1 is the same as a connection to TCP port 23 of 2001:db8::234.

A connection to TCP port 80 of 192.0.2.1 is the same as a connection to TCP port 80 of 2001:db8::5678.

Because the "all" bit is not set in 64::ff9b::f030:0 it must be assumed that there are TCP and UDP ports which cannot be reached via IPv6 so IPv4 must be used to connect to other ports.

6.4. Example 4: 'All' bit set

```
34.2.0.192.in-addr.arpa. 3600 IN AAAA 64::ff9b::f038:0
23.tcp.34.2.0.192.in-addr.arpa. 3600 IN AAAA 2001:db8::234
80.tcp.34.2.0.192.in-addr.arpa. 3600 IN AAAA 2001:db8::5678
```

Figure 7: AAAA records for example 4
Unlike example 3 (Section 6.3) the "all" bit is set in the address 64:ff9b::f038:0.

This means that there are no other connections possible via IPv4 but the connections to TCP ports 23 and 80.

7. References

7.1. Normative References


7.2. Informational References


Author’s Address

Martin D. J. Rosenau

Email: martin@rosenau-ka.de