Client IP information in DNS requests
draft-vandergaast-edns-client-ip-00

Abstract

This draft defines an EDNS0 extension to allow Authoritative Nameservers to return varying replies based upon the network address of the client that initiated the query rather than of the client’s Recursive Resolver.

Status of this Memo

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

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/1id-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 July 30, 2010.

Copyright Notice

Copyright (c) 2010 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
Table of Contents

1.  Introduction ......................................................... 3
  1.1.  Requirements notation ......................................... 3
2.  Terminology .......................................................... 4
3.  Option format ........................................................ 5
4.  Protocol description .................................................. 6
  4.1.  Querying Authoritative Nameservers ............................ 6
  4.2.  Generating a response ............................................ 7
  4.3.  Handling edns-client-ip replies and caching .................... 8
5.  IANA Considerations .................................................. 10
6.  DNSSEC Considerations ................................................ 11
7.  NAT Considerations ................................................... 12
8.  Security Considerations .............................................. 13
  8.1.  Privacy .......................................................... 13
  8.2.  Attack scenarios ................................................ 13
9.  Example ............................................................... 15
10. Acknowledgements ..................................................... 17
11. References ........................................................... 18
  11.1.  Normative References .......................................... 18
  11.2.  Informative References ........................................ 18
Authors’ Addresses ....................................................... 20
1. Introduction

Authoritative Nameservers of most major web sites today return different replies based on the perceived vicinity of the user to a particular location and knowledge of available resources. This significantly reduces the overall latency of connections established by the end user and optimizes network resource usage.

To find the best reply for a given query, most nameservers use the IP address of the incoming query to attempt to establish the location of the end user.

Most users today, however, do not query the Authoritative Nameserver directly. Instead, queries are relayed by Recursive Resolvers operated by their ISP or third parties.

When the Recursive Resolver does not use an IP address that appears to be topologically close to the end user, the results returned by those Authoritative Nameservers will be at best sub-optimal.

This draft proposes a DNS protocol extension to enable Authoritative Nameservers to return answers based on the network address of the actual client, by allowing Recursive Resolvers to include it in queries.

This draft also includes guidelines on how to best cache those results and provides recommendations on when this protocol extension should be used.

1.1. Requirements notation

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. Terminology

Authoritative Nameserver: A nameserver that has authority over one or more DNS zones. These are normally not contacted by clients directly but by Recursive Resolvers. Described in [RFC1035] chapter 6.

Recursive Resolver: A nameserver that is responsible for resolving domain names for clients by following the domain’s delegation chain, starting at the root. Recursive Resolvers frequently use caches to be able to respond to client queries quickly. Described in [RFC1035] chapter 7.

Third-party nameserver: Recursive Resolvers provided by parties that are not Internet Service Providers (ISPs). These services are often offered as substitutes for ISP-run nameservers.

Optimized reply: A reply from a nameserver that is optimized for the node that sent the request, normally based on performance (i.e. lowest latency, least number of hops, topological distance, ...).

Topologically close: Refers to two hosts being close in terms of number of hops or time it takes for a packet to travel from one host to the other. The concept of topological distance is only loosely related to the concept of geographical distance: two geographically close hosts can still be very distant from a topological perspective.
3. Option format

This draft uses an EDNS0 ([RFC2671]) option to include client IP information in DNS messages. The option is structured as follows:

```
+0 (MSB) +1 (LSB)
+-----------------------------------------------+-----------------------------------------------+
0: | OPTION-CODE |
+-----------------------------------------------+
2: | OPTION-LENGTH |
+-----------------------------------------------+
4: | FAMILY |
+-----------------------------------------------+
6: | NETMASK    | ADDRESS... |
-----------------------------------------------/
```

- (Defined in [RFC2671]) OPTION-CODE, 2 octets, for edns-client-ip is TBD.

- (Defined in [RFC2671]) OPTION-LENGTH, 2 octets, contains the length of the payload (everything after OPTION-LENGTH) in bytes.

- FAMILY, 2 octets, indicates the family of the address contained in the option, using address family codes as assigned by IANA in IANA-AFI [1].

The format of the address part depends on the value of FAMILY. This document only defines the format for FAMILY 1 (IP version 4) and 2 (IP version 6), which are as follows:

- NETMASK, 1 octet, indicating how many most-significant bits of the original ADDRESS are included (i.e. a netmask in CIDR notation).

- ADDRESS, variable number of octets, contains either an IPv4 or IPv6 address (depending on FAMILY), truncated to the number of bits indicated by the NETMASK field, with bits set to 0 to pad up to the end of the last octet used.

All fields are in network byte order.
4. Protocol description

The edns-client-ip extension allows DNS servers to propagate the network address of the client that initiated the resolution through to Authoritative Nameservers during recursion.

Servers that receive queries containing an edns-client-ip option can generate answers based on the original network address of the client. Those answers will generally be optimized for that client and other clients in the same network.

The option also allows Authoritative Nameservers to specify the network range for which the reply can be cached and re-used.

4.1. Querying Authoritative Nameservers

When a Recursive Resolver performs recursion for an incoming query, it MAY include an edns-client-ip option in queries to Authoritative Nameservers.

If an edns-client-ip option is included, the FAMILY and ADDRESS fields MUST be populated with the IP address of the client that originally initiated the resolution. The address SHOULD be truncated to an arbitrary NETMASK chosen by the owners of the Recursive Resolver as described in Section 3, following the recommendations provided in Section 8, and the NETMASK field populated accordingly.

If the incoming query originally contained an edns-client-ip option, its FAMILY, ADDRESS and NETMASK information MUST be used instead, with the only exceptions being listed in Section 8.

A Recursive Resolver set up to serve clients in private address space (as defined in [RFC1918] and [RFC4193]) MUST NOT add edns-client-ip options containing a non-public address to its queries. An edns-client-ip option MUST never contain a non-public address.

For privacy reasons, and because the whole IP address is rarely required to determine an optimized reply, the address MUST be truncated to a certain number of bits, indicated by the NETMASK field, as described in Section 8.

A Recursive Resolver MAY be configured to omit the edns-client-ip option completely when querying servers from which a delegation response is expected, for example TLD servers or root servers.
4.2. Generating a response

When a query containing an edns-client-ip option is received, an Authoritative Nameserver supporting edns-client-ip MAY use the ADDRESS and NETMASK specified in the option in order to generate an optimized reply.

Requests with an edns-client-ip option considered invalid (i.e. wrong formatting, unsupported address family, private address space) MUST be treated as if no edns-client-ip option was specified.

In the reply, the server MUST include an edns-client-ip option. The ADDRESS and FAMILY specified MUST match the ADDRESS and FAMILY in the request, while the NETMASK in the reply indicates how many bits of that ADDRESS were actually used or would have been required to calculate an optimal reply, as described below.

Note that the NETMASK value in the reply can actually be higher than the NETMASK value in the request, indicating that the address range provided in the query was not specific enough to select a single, best response, and that an optimal reply would require at least NETMASK bits of address information. The additional bits of the ADDRESS in the reply MUST be set to 0, and if octet boundaries are crossed, extra octets must be added.

Conversely, a lower NETMASK indicates that more bits than necessary were provided. In this case, the ADDRESS in the reply MUST be truncated according to the new NETMASK value, as described in Section 3.

In both cases, the value of the NETMASK in the reply has strong implications with regard to how the reply should be cached by Recursive Resolvers, as described in Section 4.3.

Servers MUST NOT include an edns-client-ip option in replies if an edns-client-ip option was not present in the request.

If the edns-client-ip option in the request is not used at all (for example if an optimized reply was temporarily unavailable or not supported for the requested domain name), a server supporting edns-client-ip MUST indicate that no bits of the ADDRESS in the request have been used by specifying a NETMASK of 0 and no ADDRESS (equivalent to the networks 0.0.0.0/0 or ::/0).

If no optimized answer could be found at all for the ADDRESS and NETMASK indicated in the query, the Authoritative Nameserver SHOULD still return the best result it knows of (i.e. by using the query source IP address instead, or a sensible default), and indicate that
this result should only be cached for the FAMILY, ADDRESS and NETMASK indicated in the request.

4.3. Handling edns-client-ip replies and caching

When a Recursive Resolver receives a reply containing an edns-client-ip option, it will return a reply to the client and cache the result as usual.

In the cache, any resource record in the answer section will be tied to the network specified by the FAMILY, ADDRESS and NETMASK fields, as detailed below.

If another query is received matching the entry in the cache, the resolver will verify that the FAMILY and ADDRESS that represent the client match any of the networks in the cache for that entry.

If the address of the client is within any of the networks in the cache, then the cached response MUST be returned as usual. In case the address of the client matches multiple networks in the cache, the entry with the highest NETMASK value MUST be returned, as with most route-matching algorithms.

If the address of the client does not match any network in the cache, then the Recursive Resolver MUST behave as if no match was found and perform resolution as usual. This is necessary to avoid sub-optimal replies in the cache from being returned to the wrong clients, and to avoid a single request coming from a client on a different network from polluting the cache with a sub-optimal reply for all the users of that resolver.

Note that every time a Recursive Resolver queries an Authoritative Nameserver by forwarding the edns-client-ip option that it received from another client, a low NETMASK in the original request could cause a sub-optimal reply to be returned by the Authoritative Nameserver.

To avoid this sub-optimal reply from being served from cache for clients for which a better reply would be available, the Recursive Resolver MUST check the NETMASK that was returned by the Authoritative Nameserver:

- If the NETMASK in the reply is higher than the NETMASK in the request, it means that the reply might be sub-optimal. A Recursive Resolver MUST return this entry from cache only to queries that do not allow a higher NETMASK to be forwarded.
o If the NETMASK in the reply is lower or equal to the NETMASK in the request, the reply is optimal, and SHOULD be returned from cache to any client in a subnet matching the ADDRESS and NETMASK indicated by the Authoritative Nameserver.

When another request is performed, the existing entries SHOULD be kept in the cache until their TTL expires, as per standard behavior.

As another reply is received, the reply will be tied to a different network. The server MAY keep in cache both replies, and return the most appropriate one depending on the address of the client.

Any reply containing an edns-client-ip option considered invalid should be treated as if no edns-client-ip option was specified at all.

Replies coming from servers not supporting edns-client-ip or otherwise not containing an edns-client-ip option SHOULD be considered as containing a NETMASK of 0 (e.g., 0.0.0.0/0 or ::/0) for all the supported families.

In any case, the response from the resolver to the client MUST NOT contain the edns-client-ip option if none was present in the client’s original request. If the original client request contained a valid edns-client-ip option that was used during recursion, the Recursive Resolver MUST include the edns-client-ip option from the Authoritative Nameserver response in the response to the client.

Enabling support for edns-client-ip in a recursive resolver will significantly increase the size of the cache, reduce the number of results that can be served from cache, and increase the load on the server. Implementing the mitigation techniques described in Section 8 is strongly recommended.
5. IANA Considerations

We request IANA to assign an option code for edns-client-ip, as specified in [RFC2671]. Within this document, the text ‘TBD’ should be replaced with the option code assigned by IANA.
6. DNSSEC Considerations

The presence or absence of an OPT resource record containing an edns-client-ip option in a DNS query does not change the usage of those resource records and mechanisms used to provide data origin authentication and data integrity to the DNS, as described in [RFC4033], [RFC4034] and [RFC4035].
7. NAT Considerations

Special awareness of edns-client-ip in devices that perform NAT as described in [RFC2663] is not required, queries can be passed through as-is. The client’s network address MUST NOT be added, and existing edns-client-ip options, if present, MUST NOT be modified by NAT devices.

Recursive Resolvers sited behind NAT devices MUST NOT add their external network address in an edns-client-ip options, and MUST behave exactly as described in the previous sections.

Note that Authoritative Nameservers or Recursive Resolvers can still provide an optimized reply by looking at the source IP of the query.
8. Security Considerations

8.1. Privacy

With the edns-client-ip option, the network address of the client that initiated the resolution becomes visible to all servers involved in the resolution process. Additionally, it will be visible from any network traversed by the DNS packets.

To protect users’ privacy, Recursive Resolvers are strongly encouraged to conceal part of the IP address of the user by truncating IPv4 addresses to 24 bits. No recommendation is provided for IPv6 at this time, but IPv6 addresses should be similarly truncated in order to not allow to uniquely identify the client.

Users who wish their full IP address to be hidden can include an edns-client-ip option specifying the wildcard address 0.0.0.0/0 (i.e. FAMILY set to 1 (IPv4), NETMASK to 0 and no ADDRESS). As described in previous sections, this option will be forwarded across all the Recursive Resolvers supporting edns-client-ip, which MUST NOT modify it to include the network address of the client.

Note that even without edns-client-ip options, any server queried directly by the user will be able to see the full client IP address. Recursive Resolvers or Authoritative Nameservers MAY use the source IP address of requests to return a cached entry or to generate an optimized reply that best matches the request.

8.2. Attack scenarios

It is simple for an arbitrary resolver or client to provide false information in the edns-client-ip option, or to send UDP packets with forged source IP addresses.

This could be used to pollute the cache of intermediate resolvers, by filling it with results that will rarely (if ever) be used, or to reverse engineer the algorithms (or data) used by the Authoritative Nameserver to calculate the optimized answers.

Even without malicious intent, third-party Recursive Resolvers providing answers to clients in multiple networks will need to cache different replies for different networks, putting significantly more pressure on the cache.

To mitigate those problems:
Recursive Resolvers implementing edns-client-ip should only enable it in deployments where it is expected to bring clear advantages to the end users. For example, when expecting clients from a variety of networks or from a wide geographical area. Due to the high cache pressure introduced by edns-client-ip, the feature must be disabled in all default configurations.

Recursive Resolvers should limit the number of networks and answers they keep in the cache for a given query.

Recursive Resolvers should limit the number of total different networks that they keep in cache.

Recursive Resolvers should never send edns-client-ip options with NETMASKS providing more bits in the ADDRESS than they are willing to cache responses for.

Recursive Resolvers should implement algorithms to improve the cache hit rate, given the size constraints indicated above. Recursive Resolvers may, for example, decide to discard more specific cache entries first.

Authoritative Nameservers and Recursive Resolvers should discard known to be wrong or known to be forged edns-client-ip options. They must at least ignore unroutable addresses, including the ones defined in [RFC1918] and [RFC4193], and should ignore and never forward edns-client-ip options specifying networks or addresses that are known not to be served by those servers when feasible.

Authoritative Nameservers consider the edns-client-ip option just as a hint to provide better results. They can decide to ignore the content of the edns-client-ip option based on black or white lists, rate limiting mechanisms, or any other logic implemented in the software.
9. Example

1. A stub resolver SR with IP address 192.0.2.37 tries to resolve www.example.com, by forwarding the query to the Recursive Resolver R from IP address IP, asking for recursion.


3. R builds a query to send to the root and .com servers. The implementation of R does not include an edns-client-ip option when querying TLD or root nameservers, because there is no expectation to receive a client-network-specific response. Thus, no edns-client-ip option is added, and resolution is performed as usual.

4. R now knows the Authoritative Nameserver ANS responsible for example.com.

5. R prepares a new query for www.example.com, including an edns-client-ip option with:
   * OPTION-CODE, set to TBD.
   * OPTION-LENGTH, set to 0x00 0x06.
   * FAMILY, set to 0x00 0x01 as IP is an IPv4 address.
   * NETMASK, set to 0x18, as it is configured to conceal the last 8 bits of every IPv4 address.
   * ADDRESS, set to 0xC0 0x00 0x02, providing only the first 24 bits of the IPv4 address.


7. Due to the internal implementation of the Authoritative Nameserver ANS, ANS finds a reply that is optimal for the whole /16 of the client that performed the request.

8. The Authoritative Nameserver ANS adds an edns-client-ip option in the reply, containing:
   * OPTION-CODE, set to TBD.
9. The Recursive Resolver R receives the reply containing an edns-client-ip option. The resolver:

* Verifies that FAMILY is set to 1, as it was in the request.
* Sees that NETMASK was lowered to 16, and truncates the IP address of stub resolver SR to that size.
* Verifies that this address matches ADDRESS in the response. If not, the option is discarded.

10. The reply is interpreted as usual. Since the reply still contains an edns-client-ip option, the ADDRESS, NETMASK, and FAMILY in the response are used to cache the entry.

11. R sends a response to stub resolver SR, without including an edns-client-ip option.

12. R receives another request to resolve www.example.com. This time, a reply is cached. The reply, however, is tied to a particular network. If the address of the client matches any network in the cache, then the reply is returned from the cache. Otherwise, another query is performed. If multiple results match, the one with the longest NETMASK is chosen, as per common best-network match algorithms.
10. Acknowledgements

The authors wish to thank the following people for reviewing early drafts of this document and for providing useful feedback: Paul S. R. Chisholm, B. Narendran, Leonidas Kontothanassis, David Presotto, Philip Rowlands, Chris Morrow, Kara Moscoe, Alex Nizhner, Warren Kumari, Richard Rabbat from Google, Terry Farmer, Mark Teodoro, Edward Lewis, Eric Burger from Neustar, David Ulevitch from OpenDNS, Patrick W. Gilmore from Akamai, Colm MacCartaigh, Richard Sheehan and all the other people that replied to our emails on various mailing lists.
11. References

11.1. Normative References


11.2. Informative References

URIs

Authors’ Addresses

Carlo Contavalli
Google
Gordon House, Barrow Street
Dublin 4
IE
Email: ccontavalli@google.com

Wilmer van der Gaast
Google
Gordon House, Barrow Street
Dublin 4
IE
Email: wilmer@google.com

Sean Leach
Neustar
46000 Center Oak Plaza
Sterling, VA 20166
VA
Email: sean.leach@neustar.com

Darryl Rodden
Neustar
46000 Center Oak Plaza
Sterling, VA 20166
VA
Email: darryl.rodden@neustar.com