                       Client Subnet in DNS Queries

Abstract

   This document describes an Extension Mechanisms for DNS (EDNS0)
   option that is in active use to carry information about the network
   that originated a DNS query and the network for which the subsequent
   response can be cached. Since it has some known operational and
   privacy shortcomings, a revision will be worked through the IETF for
   improvement.

Status of This Memo

   This document is not an Internet Standards Track specification; it is
   published for informational purposes.

   This document is a product of the Internet Engineering Task Force
   (IETF). It represents the consensus of the IETF community. It has
   received public review and has been approved for publication by the
   Internet Engineering Steering Group (IESG). Not all documents
   approved by the IESG are a candidate for any level of Internet
   Standard; see Section 2 of RFC 5741.

   Information about the current status of this document, any errata,
   and how to provide feedback on it may be obtained at
   http://www.rfc-editor.org/info/rfc7871.

Table of Contents

1. Introduction

   Many Authoritative Nameservers today return different responses based
   on the perceived topological location of the user. These servers use
   the IP address of the incoming query to identify that location.

   Since most queries come from Intermediate Recursive Resolvers, the
   source address is that of the Recursive Resolver rather than of the
   query originator.

   Traditionally, and probably still in the majority of instances,
   Recursive Resolvers are reasonably close in the topological sense to
   the Stub Resolvers or Forwarding Resolvers that are the source of
   queries. For these resolvers, using their own IP address is
   sufficient for Authoritative Nameservers that tailor responses based
   upon location of the querier.

   Increasingly, though, a class of Recursive Resolvers has arisen that
   handles query sources that are often not topologically close. The
   motivation for having such Centralized Resolvers varies but is
   usually because of some enhanced experience, such as greater cache
   security or applying policies regarding where users may connect.
   (Although political censorship usually comes to mind here, the same
   actions may be used by a parent when setting controls on where a
   minor may connect.) Similarly, many ISPs and other organizations use
   a Centralized Resolver infrastructure that can be distant from the
   clients the resolvers serve. These cases all lead to less than
   desirable responses from topology-sensitive Authoritative
   Nameservers.

   This document defines an EDNS0 [RFC6891] option to convey network
   information that is relevant to the DNS message. It will carry
   sufficient network information about the originator for the
   Authoritative Nameserver to tailor responses. It will also provide
   for the Authoritative Nameserver to indicate the scope of network
   addresses for which the tailored answer is intended. This EDNS0
   option is intended for those Recursive Resolvers and Authoritative
   Nameservers that would benefit from the extension and not for general
   purpose deployment. This is completely optional and can safely be
   ignored by servers that choose not to implement or enable it.

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

   At least a dozen different client and server implementations have
   been written based on earlier draft versions of this specification.
   The protocol is in active production use today. While the

implementations interoperate, there is varying behavior around edge
cases that were poorly specified. Known incompatibilities are
described in this document, and the authors believe that it is better
to describe the system as it is working today, even if not everyone
agrees with the details of the original specification
([VANDERGAAST]). The alternative is an undocumented and proprietary
system.

A revised proposal to improve upon the minor flaws in this protocol
will be forthcoming to the IETF.

2. Privacy Note

If we were just beginning to design this mechanism, and not
documenting existing protocol, it is unlikely that we would have done
things exactly this way.

The IETF is actively working on enhancing DNS privacy
[DPRIVE_Working_Group] and the reinjection of metadata [METADATA] has
been identified as a problematic design pattern.

As noted above however, this document primarily describes existing
behavior of a deployed method to further the understanding of the
Internet community.

We recommend that the feature be turned off by default in all
nameserver software, and that operators only enable it explicitly in
those circumstances where it provides a clear benefit for their
clients. We also encourage the deployment of means to allow users to
make use of the opt-out provided. Finally, we recommend that others
avoid techniques that may introduce additional metadata in future
work, as it may damage user trust.

Regrettably, support for the opt-out provisions of this specification
are currently limited. Only one stub resolver, getdns, is known to
be able to originate queries with anonymity requested, and as yet no
applications are known to be able to indicate that user preference to
the stub resolver.

3. 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].

4. Terminology

   ECS: EDNS Client Subnet.

   Client: A Stub Resolver, Forwarding Resolver, or Recursive Resolver.
      A client to a Recursive Resolver or a Forwarding Resolver.

   Server: A Forwarding Resolver, Recursive Resolver, or Authoritative
      Nameserver.

   Stub Resolver: A simple DNS protocol implementation on the client
      side as described in [RFC1034], Section 5.3.1. A client to a
      Recursive Resolver or a Forwarding Resolver.

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

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

   Forwarding Resolver: A nameserver that does not do iterative
      resolution itself, but instead passes that responsibility to
      another Recursive Resolver, called a "Forwarder" in [RFC2308],
      Section 1.

   Intermediate Nameserver: Any nameserver in between the Stub Resolver
      and the Authoritative Nameserver, such as a Recursive Resolver or
      a Forwarding Resolver.

   Centralized Resolvers: Intermediate Nameservers that serve a
      topologically diverse network address space.

   Tailored Response: A response from a nameserver that is customized
      for the node that sent the query, often based on performance
      (i.e., lowest latency, least number of hops, topological distance,
      etc.).

   Topologically Close: Refers to two hosts being close in terms of the
      number of hops or the 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, and two geographically distant hosts can
      be quite close on the network.

   For a more comprehensive treatment of DNS terms, please see
   [RFC7719].

5. Overview

   The general idea of this document is to provide an EDNS0 option to
   allow Recursive Resolvers, if they are willing, to forward details
   about the origin network from which a query is coming when talking to
   other nameservers.

   The format of the edns-client-subnet (ECS) EDNS0 option is described
   in Section 6 and is meant to be added in queries sent by Intermediate
   Nameservers in a way that is transparent to Stub Resolvers and end
   users, as described in Section 7.1. ECS is only defined for the
   Internet (IN) DNS class.

   As described in Section 7.2, an Authoritative Nameserver could use
   ECS as a hint to the end user's network location and provide a better
   answer. Its response would also contain an ECS option, clearly
   indicating that the server made use of this information, and that the
   answer is tied to the client's network.

   As described in Section 7.3, Intermediate Nameservers would use this
   information to cache the response.

   Some Intermediate Nameservers may also have to be able to forward ECS
   queries they receive, as described in Section 7.5.

   The mechanisms provided by ECS raise various security-related
   concerns related to cache growth, the ability to spoof EDNS0 options,
   and privacy. Section 11 explores various mitigation techniques.

   The expectation, however, is that this option will primarily be used
   between Recursive Resolvers and Authoritative Nameservers that are
   sensitive to network location issues. Most Recursive Resolvers,
   Authoritative Nameservers, and Stub Resolvers will never need to know
   about this option and will continue working as they had been.
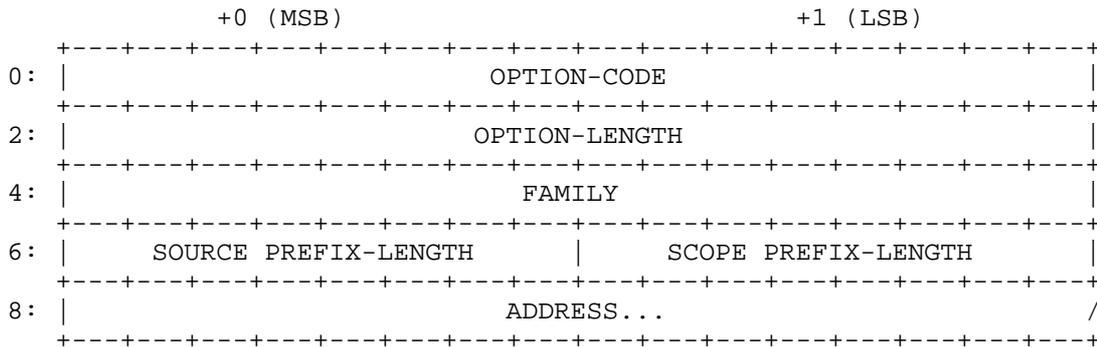
   Failure to support this option or its improper handling will, at
   worst, cause suboptimal identification of client network location,
   which is a common occurrence in current Content Delivery Network
   (CDN) setups.

Section 7.1 also provides a mechanism for Stub Resolvers to signal
Recursive Resolvers that they do not want ECS treatment for specific
queries.

Additionally, operators of Intermediate Nameservers with ECS enabled
are allowed to choose how many bits of the address of received
queries to forward or to reduce the number of bits forwarded for
queries already including an ECS option.

6. Option Format

This protocol uses an EDNS0 [RFC6891] option to include client
address information in DNS messages. The option is structured as
follows:

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

o (Defined in [RFC6891]) OPTION-CODE, 2 octets, for ECS is 8 (0x00
   0x08).

o (Defined in [RFC6891]) OPTION-LENGTH, 2 octets, contains the
   length of the payload (everything after OPTION-LENGTH) in octets.

o FAMILY, 2 octets, indicates the family of the address contained in
   the option, using address family codes as assigned by IANA in
   Address Family Numbers [Address_Family_Numbers].

The format of the address part depends on the value of FAMILY. This
document only defines the format for FAMILY 1 (IPv4) and FAMILY 2
(IPv6), which are as follows:

o SOURCE PREFIX-LENGTH, an unsigned octet representing the leftmost
   number of significant bits of ADDRESS to be used for the lookup.
   In responses, it mirrors the same value as in the queries.

   o SCOPE PREFIX-LENGTH, an unsigned octet representing the leftmost
      number of significant bits of ADDRESS that the response covers.
      In queries, it MUST be set to 0.

   o ADDRESS, variable number of octets, contains either an IPv4 or
      IPv6 address, depending on FAMILY, which MUST be truncated to the
      number of bits indicated by the SOURCE PREFIX-LENGTH field,
      padding with 0 bits to pad to the end of the last octet needed.

   o A server receiving an ECS option that uses either too few or too
      many ADDRESS octets, or that has non-zero ADDRESS bits set beyond
      SOURCE PREFIX-LENGTH, SHOULD return FORMERR to reject the packet,
      as a signal to the software developer making the request to fix
      their implementation.

   All fields are in network byte order ("big-endian", per [RFC1700],
   Data Notation).

7. Protocol Description

7.1. Originating the Option

   The ECS option should generally be added by Recursive Resolvers when
   querying Authoritative Nameservers, as described in Section 12. The
   option can also be initialized by a Stub Resolver or Forwarding
   Resolver.

7.1.1. Recursive Resolvers

   The setup of the ECS option in a Recursive Resolver depends on the
   client query that triggered the resolution process.

   In the usual case, where no ECS option was present in the client
   query, the Recursive Resolver initializes the option by setting
   FAMILY of the client's address. It then uses the value of its
   maximum cacheable prefix length to set SOURCE PREFIX-LENGTH. For
   privacy reasons, and because the whole IP address is rarely required
   to determine a tailored response, this length SHOULD be shorter than
   the full address, as described in Section 11.

   If the triggering query included an ECS option itself, it MUST be
   examined for its SOURCE PREFIX-LENGTH. The Recursive Resolver's
   outgoing query MUST then set SOURCE PREFIX-LENGTH to the shorter of
   the incoming query's SOURCE PREFIX-LENGTH or the server's maximum
   cacheable prefix length.

Finally, in both cases, SCOPE PREFIX-LENGTH is set to 0 and ADDRESS
is then added up to SOURCE PREFIX-LENGTH number of bits, with
trailing 0 bits added, if needed, to fill the final octet. The total
number of octets used MUST only be enough to cover SOURCE PREFIX-
LENGTH bits, rather than the full width that would normally be used
by addresses in FAMILY.

FAMILY and ADDRESS information MAY be used from the ECS option in the
incoming query. Passing the existing address data is supportive of
the Recursive Resolver being used as the target of a Forwarding
Resolver, but could possibly run into policy problems with regard to
usage agreements between the Recursive Resolver and Authoritative
Nameserver. See Section 12.2 for more discussion on this point. If
the Recursive Resolver will not forward FAMILY and ADDRESS data from
the incoming ECS option, it SHOULD return a REFUSED response.

Subsequent queries to refresh the data MUST, if unrestricted by an
incoming SOURCE PREFIX-LENGTH, specify the longest SOURCE PREFIX-
LENGTH that the Recursive Resolver is willing to cache, even if a
previous response indicated that a shorter prefix length was
sufficient.

7.1.2. Stub Resolvers

A Stub Resolver MAY generate DNS queries with an ECS option that sets
SOURCE PREFIX-LENGTH to limit how network information should be
revealed. An Intermediate Nameserver that receives such a query MUST
NOT make queries that include more bits of client address than in the
originating query.

A SOURCE PREFIX-LENGTH value of 0 means that the Recursive Resolver
MUST NOT add the client's address information to its queries. The
subsequent Recursive Resolver query to the Authoritative Nameserver
will then either not include an ECS option or MAY optionally include
its own address information, which is what the Authoritative
Nameserver will almost certainly use to generate any Tailored
Response in lieu of an option. This allows the answer to be handled
by the same caching mechanism as other queries, with an explicit
indicator of the applicable scope. Subsequent Stub Resolver queries
for /0 can then be answered from this cached response.

A Stub Resolver MUST set SCOPE PREFIX-LENGTH to 0. It MAY include
FAMILY and ADDRESS data, but should be prepared to handle a REFUSED
response if the Intermediate Nameserver that it queries has a policy
that denies forwarding of ADDRESS. If there is no ADDRESS set, i.e.,
SOURCE PREFIX-LENGTH is set to 0, then FAMILY SHOULD be set to the
transport over which the query is sent. This is for

interoperability; at least one major authoritative server will ignore
the option if FAMILY is not 1 or 2, even though it is irrelevant if
there are no ADDRESS bits.

7.1.3. Forwarding Resolvers

Forwarding Resolvers essentially appear to be Stub Resolvers to
whatever Recursive Resolver is ultimately handling the query, but
they look like a Recursive Resolver to their client. A Forwarding
Resolver using this option MUST prepare it as described in
Section 7.1.1, "Recursive Resolvers". In particular, a Forwarding
Resolver that implements this protocol MUST honor SOURCE PREFIX-
LENGTH restrictions indicated in the incoming query from its client.
See also Section 7.5.

Since the Recursive Resolver it contacts will treat the Forwarding
Resolver like a Stub Resolver, the Recursive Resolver's policies
regarding incoming ADDRESS information will apply in the same way.
If the Forwarding Resolver receives a REFUSED response when it sends
a query that includes a non-zero ADDRESS, it MUST retry with no
ADDRESS.

7.2. Generating a Response

7.2.1. Authoritative Nameserver

When a query containing an ECS option is received, an Authoritative
Nameserver supporting ECS MAY use the address information specified
in the option to generate a tailored response.

Authoritative Nameservers that have not implemented or enabled
support for the ECS option ought to safely ignore it within incoming
queries, per [RFC6891], Section 6.1.2. Such a server MUST NOT
include an ECS option within replies to indicate lack of support for
it. Implementers of Intermediate Nameservers should be aware,
however, that some nameservers incorrectly echo back unknown EDNS0
options. In this protocol, that should be mostly harmless, as the
SCOPE PREFIX-LENGTH should come back as 0, thus marking the response
as covering all networks.

A query with a wrongly formatted option (e.g., an unknown FAMILY)
MUST be rejected and a FORMERR response MUST be returned to the
sender, as described in [RFC6891], "Transport Considerations".

An Authoritative Nameserver that implements this protocol and
receives an ECS option MUST include an ECS option in its response to
indicate that it SHOULD be cached accordingly, regardless of whether
the client information was needed to formulate an answer. (Note that

the requirement in [RFC6891] to reserve space for the OPT record
could mean that the Answer section of the response will be truncated
and fall back to TCP indicated accordingly.) If an ECS option was
not included in a query, one MUST NOT be included in the response
even if the server is providing a Tailored Response -- presumably
based on the address from which it received the query.

FAMILY, SOURCE PREFIX-LENGTH, and ADDRESS in the response MUST match
those in the query. Echoing back these values helps to mitigate
certain attack vectors, as described in Section 11.

SCOPE PREFIX-LENGTH in the response indicates the network for which
the answer is intended.

A SCOPE PREFIX-LENGTH value longer than SOURCE PREFIX-LENGTH
indicates that the provided prefix length was not specific enough to
select the most appropriate Tailored Response. Future queries for
the name within the specified network SHOULD use the longer SCOPE
PREFIX-LENGTH. Factors affecting whether the Recursive Resolver
would use the longer length include the amount of privacy masking the
operator wants to provide their users, and the additional resource
implications for the cache.

Conversely, a shorter SCOPE PREFIX-LENGTH indicates that more bits
than necessary were provided, and the answer is suitable for a
broader range of addresses. This could be as short as 0, to indicate
that the answer is suitable for all addresses in FAMILY.

As the logical topology of any part of the network with regard to the
tailored response can vary, an Authoritative Nameserver may return
different values of SCOPE PREFIX-LENGTH for different networks.

Since some queries can result in multiple RRsets being added to the
response, there is an unfortunate ambiguity from the original
specification as to how SCOPE PREFIX-LENGTH would apply to each
individual RRset. For example, multiple types in response to an ANY
metaquery could all have different applicable SCOPE PREFIX-LENGTH
values, but this protocol only has the ability to signal one. The
response SHOULD therefore, include the longest relevant PREFIX-LENGTH
of any RRset in the answer, which could have the unfortunate side
effect of redundantly caching some data that could be cached more
broadly. For the specific case of a Canonical Name (CNAME) chain,
the Authoritative Nameserver SHOULD only place the initial CNAME
record in the Answer section, to have it cached unambiguously and
appropriately. Most modern Recursive Resolvers restart the query
with the CNAME, so the remainder of the chain is typically ignored

anyway. For message-focused resolvers, rather than RRset-focused
ones, this will mean caching the entire CNAME chain at the longest
PREFIX-LENGTH of any RRset in the chain.

The specific logic that an Authoritative Nameserver uses to choose a
tailored response is not in the scope of this document. Implementers
are encouraged, however, to carefully consider their selection of
SCOPE PREFIX-LENGTH for the response in the event that the best
tailored response cannot be determined, and what the implications
would be over the life of the TTL.

Authoritative Nameservers might have situations where one Tailored
Response is appropriate for a relatively broad address range, such as
an IPv4 /20, except for some exceptions, such as a few /24 ranges
within that /20. Because it can't be guaranteed that queries for all
longer prefix lengths would arrive before one that would be answered
by the shorter prefix length, an Authoritative Nameserver MUST NOT
overlap prefixes.

When the Authoritative Nameserver has a longer prefix length Tailored
Response within a shorter prefix length Tailored Response, then
implementations can either:

1. Deaggregate the shorter prefix response into multiple longer
    prefix responses, or

2. Alert the operator that the order of queries will determine which
    answers get cached, and either warn and continue or treat this as
    an error and refuse to load the configuration.

This choice should be documented for the operator, for example, in
the user manual.

When deaggregating to correct the overlap, prefix lengths should be
optimized to use the minimum necessary to cover the address space, in
order to reduce the overhead that results from having multiple copies
of the same answer. As a trivial example, if the Tailored Response
for 1.2.0/20 is A but there is one exception of 1.2.3/24 for B, then
the Authoritative Nameserver would need to provide Tailored Responses
for 1.2.0/23, 1.2.2/24, 1.2.4/22, and 1.2.8/21 all pointing to A, and
1.2.3/24 to B.

7.2.2. Intermediate Nameserver

When an Intermediate Nameserver uses ECS, whether it passes an ECS
option in its own response to its client is predicated on whether the
client originally included the option. Because a client that did not
use an ECS option might not be able to understand it, the server MUST

NOT provide one in its response. If the client query did include the
option, the server MUST include one in its response, especially as it
could be talking to a Forwarding Resolver, which would need the
information for its own caching.

If an Intermediate Nameserver receives a response that has a longer
SCOPE PREFIX-LENGTH than SOURCE PREFIX-LENGTH that it provided in its
query, it SHOULD still provide the result as the answer to the
triggering client request even if the client is in a different
address range. The Intermediate Nameserver MAY instead opt to retry
with a longer SOURCE PREFIX-LENGTH to get a better reply before
responding to its client, as long as it does not exceed a SOURCE
PREFIX-LENGTH specified in the query that triggered resolution, but
this obviously has implications for the latency of the overall
lookup.

The logic for using the cache to determine whether the Intermediate
Nameserver already knows the response to provide to its client is
covered in the next section.

7.3. Handling ECS Responses and Caching

When an Intermediate Nameserver receives a response containing an ECS
option and without the TC bit set, it SHOULD cache the result based
on the data in the option. If the TC bit was set, the Intermediate
Resolver SHOULD retry the query over TCP to get the complete Answer
section for caching.

If FAMILY, SOURCE PREFIX-LENGTH, and SOURCE PREFIX-LENGTH bits of
ADDRESS in the response don't match the non-zero fields in the
corresponding query, the full response MUST be dropped, as described
in Section 11. In a response to a query that specified only SOURCE
PREFIX-LENGTH for privacy masking, the FAMILY and ADDRESS fields MUST
contain the appropriate non-zero information that the Authoritative
Nameserver used to generate the answer, so that it can be cached
accordingly.

If no ECS option is contained in the response, the Intermediate
Nameserver SHOULD treat this as being equivalent to having received a
SCOPE PREFIX-LENGTH of 0, which is an answer suitable for all client
addresses. See further discussion on the security implications of
this in Section 11.

If a REFUSED response is received from an Authoritative Nameserver,
an ECS-aware resolver MUST retry the query without ECS to distinguish
the response from one where the Authoritative Nameserver is not
responsible for the name, which is a common convention for the
REFUSED status. Similarly, a client of a Recursive Resolver SHOULD

retry after receiving a REFUSED response because it is not
sufficiently clear whether the REFUSED response was because of the
ECS option or some other reason.

7.3.1. Caching the Response

   In the cache, all resource records in the Answer section MUST be tied
   to the network specified in the response. The appropriate prefix
   length depends on the relationship between SOURCE PREFIX-LENGTH,
   SCOPE PREFIX-LENGTH, and the maximum cacheable prefix length
   configured for the cache.

   If SCOPE PREFIX-LENGTH is not longer than SOURCE PREFIX-LENGTH, store
   SCOPE PREFIX-LENGTH bits of ADDRESS, and then mark the response as
   valid for all addresses that fall within that range.

   Similarly, if SOURCE PREFIX-LENGTH is the maximum configured for the
   cache, store SOURCE PREFIX-LENGTH bits of ADDRESS, and then mark the
   response as valid for all addresses that fall within that range.

   If SOURCE PREFIX-LENGTH is shorter than the configured maximum and
   SCOPE PREFIX-LENGTH is longer than SOURCE PREFIX-LENGTH, store SOURCE
   PREFIX-LENGTH bits of ADDRESS, and then mark the response as valid
   only to answer client queries that specify exactly the same SOURCE
   PREFIX-LENGTH in their own ECS option.

   The handling of DNSSEC-related records in the Answer section was
   unspecified in the original draft version of this document and is
   inconsistently handled in existing implementations. A Resource
   Record Signature (RRSIG) must obviously be tied to the RRset that it
   signs, but it is RECOMMENDED that all other DNSSEC records be scoped
   at /0. See Section 9 for more information.

   Note that the Additional and Authority sections from a DNS response
   message are specifically excluded here. Any records from these
   sections MUST NOT be tied to a network. See Section 7.4 for more
   information.

   Records that are cached as /0 because of a query's SOURCE PREFIX-
   LENGTH of 0 MUST be distinguished from those that are cached as /0
   because of a response's SCOPE PREFIX-LENGTH of 0. The former should
   only be used for other /0 queries that the Intermediate Resolver
   receives, but the latter is suitable as a response for all networks.

   Although omitting network-specific caching will significantly
   simplify an implementation, the resulting drop in cache hits is very
   likely to defeat most latency benefits provided by ECS. Therefore,
   implementing full caching support as described in this section is
   strongly RECOMMENDED.

   Enabling support for ECS in an Intermediate Nameserver 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 11 is strongly recommended. For cache size issues,
   implementers should consider data storage formats that allow the same
   answer data to be shared among multiple prefixes.

7.3.2. Answering from Cache

   Cache lookups are first done as usual for a DNS query, using the
   query tuple of <name, type, class>. Then, the appropriate RRset MUST
   be chosen based on the longest prefix matching. The client address
   to use for comparison will depend on whether the Intermediate
   Nameserver received an ECS option in its client query.

   o If no ECS option was provided, the client's address is used.

   o If there was an ECS option specifying SOURCE PREFIX-LENGTH and
      ADDRESS covering the client's address, the client address is used
      but SOURCE PREFIX-LENGTH is initially ignored. If no covering
      entry is found and SOURCE PREFIX-LENGTH is shorter than the
      configured maximum length allowed for the cache, repeat the cache
      lookup for an entry that exactly matches SOURCE PREFIX-LENGTH.
      These special entries, which do not cover longer prefix lengths,
      occur as described in the previous section.

   o If there was an ECS option with an ADDRESS, the ADDRESS from it
      MAY be used if the local policy allows. The policy can vary
      depending on the agreements the operator of the Intermediate
      Nameserver has with Authoritative Nameserver operators; see
      Section 12.2. If the policy does not allow it, a REFUSED response
      SHOULD be sent. See Section 7.5 for more information.

   If a matching network is found and the relevant data is unexpired,
   the response is generated as per Section 7.2.

   If no matching network is found, the Intermediate Nameserver MUST
   perform resolution as usual. This is necessary to avoid Tailored
   Responses in the cache from being returned to the wrong clients, and

to avoid a single query coming from a client on a different network
from polluting the cache with a Tailored Response for all the users
of that resolver.

7.4. Delegations and Negative Answers

The prohibition against tying ECS data to records from the Authority
and Additional sections left an unfortunate ambiguity in the original
specification, primarily with regard to negative answers. The
expectation of the original authors was that ECS would only really be
used for address requests and the positive result in the response's
Answer section, which was the use case that was driving the
definition of the protocol.

For negative answers, some independent implementations of both
resolvers and authorities did not see the section restriction as
necessarily meaning that a given name and type must only have either
positive ECS-tagged answers or a negative answer. They support being
able to tell one part of the network that the data does not exist,
while telling another part of the network that it does.

Several other implementations, however, do not support being able to
mix positive and negative answers; thus, interoperability is a
problem. It is RECOMMENDED that no specific behavior regarding
negative answers be relied upon, but that Authoritative Nameservers
should conservatively expect that Intermediate Nameservers will treat
all negative answers as /0; therefore, they SHOULD set SCOPE PREFIX-
LENGTH accordingly.

This issue is expected to be revisited in a future revision of the
protocol, possibly blessing the mixing of positive and negative
answers. There are implications for cache data structures that
developers should consider when writing new ECS code.

The delegations case is a bit easier to tease out. In operational
practice, if an authoritative server is using address information to
provide customized delegations, it is the resolver that will be using
the answer for its next iterative query. Addresses in the Additional
section SHOULD therefore ignore ECS data, and the Authoritative
Nameserver SHOULD return a zero SCOPE PREFIX-LENGTH on delegations.
A Recursive Resolver SHOULD treat a non-zero SCOPE PREFIX LENGTH in a
delegation as though it were zero.

7.5. Transitivity

   Generally, ECS options will only be present in DNS messages between a
   Recursive Resolver and an Authoritative Nameserver, i.e., one hop.
   However, in certain configurations, for example, multi-tier
   nameserver setups, it may be necessary to implement transitive
   behavior on Intermediate Nameservers.

   Any Intermediate Nameserver that forwards ECS options received from
   its clients MUST fully implement the caching behavior described in
   Section 7.3.

   An Intermediate Nameserver MAY forward ECS options with address
   information. This information MAY match the source IP address of the
   incoming query, and MAY have more or fewer address bits than the
   nameserver would normally include in a locally originated ECS option.
   If an Intermediate Nameserver receives a query with SOURCE PREFIX-
   LENGTH set to 0, it MUST NOT include client address information in
   queries made to resolve that client's request (see Section 7.1.2).

   If, for any reason, the Intermediate Nameserver does not want to use
   the information in an ECS option it receives (too little address
   information, network address from a range not authorized to use the
   server, private/unroutable address space, etc.), it SHOULD drop the
   query and return a REFUSED response. Note again that a query MUST
   NOT be refused solely because it provides 0 address bits.

   Be aware that at least one major existing implementation does not
   return REFUSED and instead just processes the query as though the
   problematic information were not present. This can lead to anomalous
   situations, such as a response from the Intermediate Nameserver that
   indicates it is tailored for one network (the one passed in the
   original query, since the ADDRESS must match) when actually it is for
   another network (the one which contains the address that the
   Intermediate Nameserver saw as making the query).

8. IANA Considerations

   IANA has assigned option code 8 in the "DNS EDNS0 Option Codes (OPT)"
   registry to edns-client-subnet.

   IANA has updated the reference to refer to this RFC.

9. DNSSEC Considerations

   The presence or absence of an EDNS0 OPT resource record ([RFC6891])
   containing an ECS option in a DNS query does not change the usage of
   the resource records and mechanisms used to provide data origin
   authentication and data integrity to the DNS, as described in
   [RFC4033], [RFC4034], and [RFC4035]. OPT records are not signed.

   Use of this option, however, does imply increased DNS traffic between
   any given Recursive Resolver and Authoritative Nameserver, which
   could be another barrier to further DNSSEC adoption in this area.

   The initial version of this protocol, against which several
   Authoritative and Recursive Nameserver implementations were written,
   did not discuss the handling of DNSSEC RRs; thus, it is expected that
   there are operational inconsistencies in handling them.

   Given the intention of this document to describe how ECS is currently
   deployed, specifying new requirements for DNSSEC handling is out of
   scope. However, some recommendations can be made as to what is most
   likely to result in successful interoperation for a DNSSEC-signed ECS
   zone, mainly from the point of view of Authoritative Nameservers.

   Most DNSSEC records SHOULD be scoped at /0, except for the RRSIG
   records, which MUST be tied to the RRset that they sign in a Tailored
   Response. While it is possible to conceive of a way to get other
   DNSSEC records working in a network-specific way, it has little
   apparent benefit or likelihood of working with deployed validating
   resolvers.

   One further implication here is that, despite the discussion about
   negative answers in Section 7.4, scoping NextSECure (NSEC) or NSEC3
   records at /0 per the previous paragraph necessarily implies that
   DNSSEC-signed negative answers must also be network-invariant.

10. NAT Considerations

   Special awareness of ECS in devices that perform Network Address
   Translation (NAT) as described in [RFC2663] is not required; queries
   can be passed through as is. The client's network address SHOULD NOT
   be added, and existing ECS options, if present, SHOULD NOT be
   modified by NAT devices.

   In large-scale global networks behind a NAT device (but, for example
   with Centralized Resolver infrastructure), an internal Intermediate
   Nameserver might have detailed network layout information, and may

know which external subnets are used for egress traffic by each
internal network. In such cases, the Intermediate Nameserver MAY use
that information when originating ECS options.

In other cases, if a Recursive Resolver knows that it is situated
behind a NAT device, it SHOULD NOT originate ECS options with their
external IP address and instead rely on downstream Intermediate
Nameservers to do so. It MAY, however, choose to include the option
with their internal address for the purposes of signaling its own
limit for SOURCE PREFIX-LENGTH.

Full treatment of special network addresses is beyond the scope of
this document; handling them will likely differ according to the
operational environments of each service provider. As a general
guideline, if an Authoritative Nameserver on the publicly routed
Internet receives a query that specifies an ADDRESS in [RFC1918] or
[RFC4193] private address space, it SHOULD ignore ADDRESS and look up
its answer based on the address of the Recursive Resolver. In the
response, it SHOULD set SCOPE PREFIX-LENGTH to cover all of the
relevant private space. For example, a query for ADDRESS 10.1.2.0
with a SOURCE PREFIX-LENGTH of 24 would get a returned SCOPE PREFIX-
LENGTH of 8. The Intermediate Nameserver MAY elect to cache the
answer under one entry for special-purpose addresses [RFC6890]; see
Section 11.3 of this document.

11. Security Considerations

11.1. Privacy

With the ECS 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 user's IP address by truncating
IPv4 addresses to 24 bits. 56 bits are recommended for IPv6, based on
[RFC6177].

ISPs should have more detailed knowledge of their own networks. That
is, they might know that all 24-bit prefixes in a /20 are in the same
area. In those cases, for optimal cache utilization and improved
privacy, the ISP's Recursive Resolver SHOULD truncate IP addresses in
this /20 to just 20 bits, instead of 24 as recommended above.

Users who wish their full IP address to be hidden need to configure
their client software, if possible, to include an ECS option
specifying the wildcard address (i.e., a SOURCE PREFIX-LENGTH of 0).

As described in previous sections, this option will be forwarded across all the Recursive Resolvers supporting ECS, which MUST NOT modify it to include the network address of the client.

Note that even without an ECS option, 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 queries to return a cached entry or to generate a Tailored Response that best matches the query.

11.2. Birthday Attacks

ECS adds information to the DNS query tuple (q-tuple). This allows an attacker to send a caching Intermediate Nameserver multiple queries with spoofed IP addresses either in the ECS option or as the source IP. These queries will trigger multiple outgoing queries with the same name, type, and class, just with different address information in the ECS option.

With multiple queries for the same name in flight, the attacker has a higher chance of success to send a matching response with SCOPE PREFIX-LENGTH set to 0 to get it cached for all hosts.

To counter this, the ECS option in a response packet MUST contain the full FAMILY, ADDRESS, and SOURCE PREFIX-LENGTH fields from the corresponding query. Intermediate Nameservers processing a response MUST verify that these match, and they SHOULD discard the entire response if they do not.

The requirement to discard is categorized as "SHOULD" instead of "MUST" because it stands in opposition to the instruction in Section 7.3, which states that a response lacking an ECS option should be treated as though it had one of SCOPE PREFIX-LENGTH of 0. If that is always true, then an attacker does not need to worry about matching the original ECS option data and just needs to flood back responses that have no ECS option at all.

This type of attack could be detected in ongoing operations by marking whether the responding nameserver had previously been sending ECS options and/or by taking note of an incoming flood of bogus responses and flagging the relevant query for re-resolution. This type of detection is more complex than existing nameserver responses to spoof floods, and it would also need to be sensitive to a nameserver legitimately stopping ECS replies even though it had previously given them.

11.3. Cache Pollution

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

   This could be used to:

   o pollute the cache of Intermediate Resolvers by filling it with
      results that will rarely (if ever) be used.

   o reverse-engineer the algorithms (or data) used by the
      Authoritative Nameserver to calculate Tailored Responses.

   o mount a denial-of-service attack against an Intermediate
      Nameserver by forcing it to perform many more recursive queries
      than it would normally do, due to how caching is handled for
      queries containing the ECS option.

   Even without malicious intent, Centralized Resolvers providing
   answers to clients in multiple networks will need to cache different
   responses for different networks, putting more memory pressure on the
   cache.

   To mitigate those problems:

   o Recursive Resolvers implementing ECS should only enable it in
      deployments where it is expected to bring clear advantages to the
      end users, such as when expecting clients from a variety of
      networks or from a wide geographical area. Due to the high cache
      pressure introduced by ECS, the feature SHOULD be disabled in all
      default configurations.

   o Recursive Resolvers SHOULD limit the number of networks and
      answers they keep in the cache for any given query.

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

   o Recursive Resolvers MUST NOT send an ECS option with SOURCE
      PREFIX-LENGTH providing more bits in ADDRESS than they are willing
      to cache responses for.

   o 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.

   o Authoritative Nameservers and Recursive Resolvers should discard
      ECS options that are either obviously forged or otherwise known to
      be wrong. They SHOULD at least treat unroutable addresses, such
      as some of the address blocks defined in [RFC6890], as equivalent
      to the Recursive Resolver's own identity. They SHOULD ignore and
      never forward ECS options specifying other routable addresses that
      are known not to be served by the query source.

   o The ECS option is just a hint to Authoritative Nameservers for
      customizing results. They can decide to ignore the content of the
      ECS option based on blacklists or whitelists, rate-limiting
      mechanisms, or any other logic implemented in the software.

12. Sending the Option

   When implementing a Recursive Resolver, there are two strategies on
   deciding when to include an ECS option in a query. At this stage,
   it's not clear which strategy is best.

12.1. Probing

   A Recursive Resolver can send the ECS option with every outgoing
   query. However, it is RECOMMENDED that resolvers remember which
   Authoritative Nameservers did not return the option with their
   response and omit client address information from subsequent queries
   to those nameservers.

   Additionally, Recursive Resolvers SHOULD be configured never to send
   the option when querying root, top-level, and effective top-level
   (i.e., "public suffix" [Public_Suffix_List]) domain servers. These
   domains are delegation-centric and are very unlikely to generate
   different responses based on the address of the client.

   When probing, it is important that several things are probed: support
   for ECS, support for EDNS0, support for EDNS0 options, or possibly an
   unreachable nameserver. Various implementations are known to drop
   DNS packets with OPT RRs (with or without options), thus several
   probes are required to discover what is supported.

   Probing, if implemented, MUST be repeated periodically, e.g., daily.
   If an Authoritative Nameserver indicates ECS support for one zone, it
   is to be expected that the nameserver supports ECS for all of its
   zones. Likewise, an Authoritative Nameserver that uses ECS
   information for one of its zones MUST indicate support for the option
   in all of its responses to ECS queries. If the option is supported
   but not actually used for generating a response, its SCOPE PREFIX-
   LENGTH MUST be set to 0.

12.2. Whitelist

   As described previously, it is expected that only a few Recursive
   Resolvers will need to use ECS, and that it will generally be enabled
   only if it offers a clear benefit to the users.

   To avoid the complexity of implementing a probing and detection
   mechanism (and the possible query loss/delay that may come with it),
   an implementation could use a whitelist of Authoritative Nameservers
   to send the option to, likely specified by their domain name.
   Implementations MAY also allow additional configuring of this based
   on other criteria, such as zone or query type. As of the time of
   this writing, at least one implementation makes use of a whitelist.

   An advantage of using a whitelist is that partial client address
   information is only disclosed to nameservers that are known to use
   the information, improving privacy.

   A drawback is scalability. The operator needs to track which
   Authoritative Nameservers support ECS, making it harder for new
   Authoritative Nameservers to start using the option.

   Similarly, Authoritative Nameservers can also use whitelists to limit
   the feature to only certain clients. For example, a CDN that does
   not want all of their mapping trivially walked might require a legal
   agreement with the Recursive Resolver operator, to clearly describe
   the acceptable use of the feature.

   The maintenance of access control mechanisms is out of scope for this
   protocol definition.

13. Example

   1.   A Stub Resolver, SR, with the IP address
        2001:0db8:fd13:4231:2112:8a2e:c37b:7334 tries to resolve
        www.example.com by forwarding the query to the Recursive
        Resolver, RNS, asking for recursion.

   2.   RNS, supporting ECS, looks up www.example.com in its cache. An
        entry is found neither for www.example.com nor for example.com.

   3.   RNS builds a query to send to the root and .com servers. The
        implementation of RNS provides facilities so that an
        administrator can configure it not to forward ECS in certain
        cases. In particular, RNS is configured not to include an ECS
        option when talking to Top-Level-Domain or root nameservers, as
        described in Section 7.1. Thus, no ECS option is added, and
        resolution is performed as usual.

4. RNS now knows the next server to query: the Authoritative
    Nameserver, ANS, responsible for example.com.

5. RNS prepares a new query for www.example.com, including an ECS
    option with:

    * OPTION-CODE set to 8.

    * OPTION-LENGTH set to 0x00 0x0b for the following fixed 4
       octets plus the 7 octets that will be used for ADDRESS.

    * FAMILY set to 0x00 0x02, as IP is an IPv6 address.

    * SOURCE PREFIX-LENGTH set to 0x38, as RNS is configured to
       conceal the last 72 bits of every IPv6 address.

    * SCOPE PREFIX-LENGTH set to 0x00, as specified by this
       document for all queries.

    * ADDRESS set to 0x20 0x01 0x0d 0xb8 0xfd 0x13 0x42, providing
       only the first 56 bits of the IPv6 address.

6. The query is sent. ANS understands and uses ECS. It parses the
    ECS option, and generates a Tailored Response.

7. Due its internal implementation, ANS finds a response that is
    tailored for the whole /16 of the client that performed the
    query.

8. ANS adds an ECS option in the response, containing:

    * OPTION-CODE set to 8.

    * OPTION-LENGTH set to 0x00 0x07.

    * FAMILY set to 0x00 0x02.

    * SOURCE PREFIX-LENGTH set to 0x38, copied from the query.

    * SCOPE PREFIX-LENGTH set to 0x30, indicating a /48 network.

    * ADDRESS set to 0x20 0x01 0x0d 0xb8 0xfd 0x13 0x42, copied
       from the query.

9. RNS receives the response containing an ECS option. It verifies
    that FAMILY, SOURCE PREFIX-LENGTH, and ADDRESS match the query.
    If not, the message is discarded.

10. The response is interpreted as usual. Since the response
     contains an ECS option, ADDRESS, SCOPE PREFIX-LENGTH, and FAMILY
     in the response are used to cache the entry.

11. RNS sends a response to Stub Resolver, SR, without including an
     ECS option.

12. RNS receives another query to resolve www.example.com. This
     time, a response is cached. The response, however, is tied to a
     particular network. If the client's address matches any network
     in the cache, then the response is returned from the cache.
     Otherwise, another query is performed. If multiple results
     match, the one with the longest SCOPE PREFIX-LENGTH is chosen,
     as per common best-network-match algorithms.

14. References

14.1. Normative References

   [RFC1034] Mockapetris, P., "Domain Names - Concepts and Facilities",
              STD 13, RFC 1034, DOI 10.17487/RFC1034, November 1987,
              <http://www.rfc-editor.org/info/rfc1034>.

   [RFC1035] Mockapetris, P., "Domain Names - Implementation and
              Specification", STD 13, RFC 1035, DOI 10.17487/RFC1035,
              November 1987, <http://www.rfc-editor.org/info/rfc1035>.

   [RFC1700] Reynolds, J. and J. Postel, "Assigned Numbers", RFC 1700,
              DOI 10.17487/RFC1700, October 1994,
              <http://www.rfc-editor.org/info/rfc1700>.

   [RFC1918] Rekhter, Y., Moskowitz, B., Karrenberg, D., de Groot, G.,
              and E. Lear, "Address Allocation for Private Internets",
              BCP 5, RFC 1918, DOI 10.17487/RFC1918, February 1996,
              <http://www.rfc-editor.org/info/rfc1918>.

   [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
              Requirement Levels", BCP 14, RFC 2119,
              DOI 10.17487/RFC2119, March 1997,
              <http://www.rfc-editor.org/info/rfc2119>.

   [RFC4033] Arends, R., Austein, R., Larson, M., Massey, D., and S.
              Rose, "DNS Security Introduction and Requirements",
              RFC 4033, DOI 10.17487/RFC4033, March 2005,
              <http://www.rfc-editor.org/info/rfc4033>.

   [RFC4034] Arends, R., Austein, R., Larson, M., Massey, D., and S.
              Rose, "Resource Records for the DNS Security Extensions",
              RFC 4034, DOI 10.17487/RFC4034, March 2005,
              <http://www.rfc-editor.org/info/rfc4034>.

   [RFC4035] Arends, R., Austein, R., Larson, M., Massey, D., and S.
              Rose, "Protocol Modifications for the DNS Security
              Extensions", RFC 4035, DOI 10.17487/RFC4035, March 2005,
              <http://www.rfc-editor.org/info/rfc4035>.

   [RFC4193] Hinden, R. and B. Haberman, "Unique Local IPv6 Unicast
              Addresses", RFC 4193, DOI 10.17487/RFC4193, October 2005,
              <http://www.rfc-editor.org/info/rfc4193>.

   [RFC6177] Narten, T., Huston, G., and L. Roberts, "IPv6 Address
              Assignment to End Sites", BCP 157, RFC 6177,
              DOI 10.17487/RFC6177, March 2011,
              <http://www.rfc-editor.org/info/rfc6177>.

   [RFC6890] Cotton, M., Vegoda, L., Bonica, R., Ed., and B. Haberman,
              "Special-Purpose IP Address Registries", BCP 153,
              RFC 6890, DOI 10.17487/RFC6890, April 2013,
              <http://www.rfc-editor.org/info/rfc6890>.

   [RFC6891] Damas, J., Graff, M., and P. Vixie, "Extension Mechanisms
              for DNS (EDNS(0))", STD 75, RFC 6891,
              DOI 10.17487/RFC6891, April 2013,
              <http://www.rfc-editor.org/info/rfc6891>.

14.2. Informative References

   [Address_Family_Numbers]
              IANA, "Address Family Numbers",
              <http://www.iana.org/assignments/address-family-numbers>.

   [DPRIVE_Working_Group]
              IETF, "PNS PRIVate Exchange (dprive) DPRIVE Working
              Group", 2015,
              <https://datatracker.ietf.org/wg/dprive/charter/>.

   [METADATA]
              Hardie, T., Ed., "Design considerations for Metadata
              Insertion", Work in Progress, draft-hardie-privsec-
              metadata-insertion-02, March 2016.

   [Public_Suffix_List]
              "Public Suffix List", <https://publicsuffix.org/>.

   [RFC2308] Andrews, M., "Negative Caching of DNS Queries (DNS
              NCACHE)", RFC 2308, DOI 10.17487/RFC2308, March 1998,
              <http://www.rfc-editor.org/info/rfc2308>.

   [RFC2663] Srisuresh, P. and M. Holdrege, "IP Network Address
              Translator (NAT) Terminology and Considerations",
              RFC 2663, DOI 10.17487/RFC2663, August 1999,
              <http://www.rfc-editor.org/info/rfc2663>.

   [RFC7719] Hoffman, P., Sullivan, A., and K. Fujiwara, "DNS
              Terminology", RFC 7719, DOI 10.17487/RFC7719, December
              2015, <http://www.rfc-editor.org/info/rfc7719>.

   [VANDERGAAST]
              Contavalli, C., Gaast, W., Leach, S., and E. Lewis,
              "Client Subnet in DNS Requests", Work in Progress,
              draft-vandergaast-edns-client-subnet-02, July 2013.

Contributors

   The individuals below contributed significantly to this document.

   Edward Lewis
   ICANN
   12025 Waterfront Drive, Suite 300
   Los Angeles, CA 90094-2536
   United States

   Email: edward.lewis@icann.org

   Sean Leach
   Fastly
   P.O. Box 78266
   San Francisco, CA 94107
   United States

   Jason Moreau
   Akamai Technologies
   150 Broadway
   Cambridge, MA 02142-1413
   United States

Authors' Addresses

   Carlo Contavalli
   Google
   1600 Amphitheater Parkway
   Mountain View, CA 94043
   United States

   Email: ccontavalli@google.com

   Wilmer van der Gaast
   Google
   Belgrave House, 76 Buckingham Palace Road
   London SW1W 9TQ
   United Kingdom

   Email: wilmer@google.com

   David C Lawrence
   Akamai Technologies
   150 Broadway
   Cambridge, MA 02142-1054
   United States

   Email: tale@akamai.com

   Warren Kumari
   Google
   1600 Amphitheatre Parkway
   Mountain View, CA 94043
   United States

   Email: warren@kumari.net