Abstract

This document talks about SMTP operational experiences in IPv4/v6 dual stack environments. As IPv6-capable SMTP servers are deployed, it has become apparent that certain configurations are necessary in IPv6-capable MX DNS records for stable dual-stack (IPv4 and IPv6) SMTP operation. This document clarifies the problems that exist in the transition period between IPv4 SMTP and IPv6 SMTP. It also defines operational requirements for stable IPv4/v6 SMTP operation.

This document does not define any new protocol.

1. Introduction

Deliveries of mail messages to the final mail drop is not always done by direct IP communication with submiter and final receiver, and there may be some intermediate hosts to relay the messages. So it is difficult to
know at message submission (also at receiver side) that all intermediate relay hosts are properly configured. It is not so easy to configure all the system with consistency since mail message delivery system is rather complex on DNS setting than other Internet services. For the transition state from IPv4 to IPv6, both IPv4 and IPv6 interoperability should be kept more carefully.

There are several technologies defined for the transition from IPv4 to IPv6. This document concentrates on SMTP issues in a dual-stack environment. Afterall, there are no special SMTP considerations for translators; If there is SMTP traffic from an IPv6 MTA to an IPv4 MTA over an IPv6-to-IPv4 translator, the IPv4 MTA will consider this normal IPv4 SMTP traffic. Protocols like IDENT [Johns, 1993], however, may require special consideration when translators are used.

The following sections explain how to make IPv4 SMTP and IPv6 SMTP coexist in a dual-stack environment.

This document does not discuss the problems encountered when the sending MTA and the receiving MTA have no common protocol (e.g. the sending MTA is IPv4-only while the receiving MTA is IPv6-only). Such a situation should be resolved by making either side dual-stack or by making either side use a protocol translator.

2. Basic DNS resource record definitions for mail routing

Mail messages on the Internet are delivered based on domain name system generally. MX RRs are looked up to know destination hosts associated with domain part of a mail address. Similar to the way RFC’s for IPv6 DNS lookup [Thomson, 1995] use IN class for both IPv4 and IPv6, IN MX records will be used for both IPv4 and IPv6 on mail message routing, hosts which have IPv6 transport and want to be delivered with the IPv6 transport must define IPv6 IP addresses for the host name as well as IPv4 IP addresses.

A MX RR have two data, a preference value and the name of destination host. IP addresses for the destination host are also looked up to make SMTP transport [Partridge, 1986]. In IPv4 environment, IPv4 IP addresses are defined with A RRs.

For example, IPv6 only site may have the following DNS definitions:

```
example.org.           IN MX  1  mx1.example.org.
                       IN MX  10 mx10.example.org.
mx1.example.org.        IN AAAA 3ffe:501:ffff::1
mx10.example.org.       IN AAAA 3ffe:501:ffff::2
```

In transition period from IPv4 to IPv6, there are many IPv4 sites, and such sites will not have mail interoperability with IPv6 only sites. For the transition period, every IPv6 sites should have both transport for each domain part of mail addresses, e.g, for example:
example.org. IN MX 1 mx1.example.org.
IN MX 10 mx10.example.org.
mx1.example.org. IN A 192.0.2.1
IN AAAA 3ffe:501:ffff::1
mx10.example.org. IN AAAA 3ffe:501:ffff::2
IN A 192.0.2.2

But, every host may not support dual stack operation, some host entries
may have only IPv4 or IPv6 RRs:

example.org. IN MX 1 mx1.example.org.
IN MX 10 mx10.example.org.
mx1.example.org. IN AAAA 3ffe:501:ffff::1
mx10.example.org. IN A 192.0.2.1

In the following sections, how sender side operates with IPv4/IPv6
combined RR definitions (section 3), and how receiver side should define
RRs to keep interoperability with both IPv4 and IPv6 Internet (section
4) are considered.

3. SMTP sender algorithm in a dual-stack environment

In a dual-stack environment MX records for a domain resemble the
following:

example.org. IN MX 1 mx1.example.org.
IN MX 10 mx10.example.org.
mx1.example.org. IN A 192.0.2.1 ; dual-stack
IN AAAA 3ffe:501:ffff::1
mx10.example.org. IN AAAA 3ffe:501:ffff::2 ; IPv6 only

For a single MX record there are many possible final states, including:
(a) one or more A records for the IPv4 destination, (b) one or more AAAA
records for the IPv6 destination, (c) a mixture of A and AAAA records.
Because multiple MX records may be defined using different preference
values, multiple addresses based on multiple MX’s must be traversed.
Domains without MX records and failure recovery cases must be handled
properly as well.

The algorithm for an SMTP sender is basically the same as that for an
IPv4-only sender, but it now includes AAAA lookups of MX records for
SMTP-over-IPv6 delivery. IPv4/v6 dual stack destinations should be
treated just like multihomed destinations as described in RFC2821
[Klensin, 2001] section 5. When there is no reachable destination
address record found (for example, the sender MTA is IPv4 only and there
are no A records available) the case should be treated just like MX
records without address records, and deliveries never fail because of no
known address if other addresses are available related to other MX
records.
; if the sender MTA is IPv4 only, email delivery to a.example.org
; should fail with the same error as deliveries to b.example.org.
a.example.org. IN MX 1 mx1.a.example.org.
mx1.a.example.org. IN AAAA 3ffe:501:ffff::1 ; IPv6 only
b.example.org. IN MX 1 mx1.b.example.org.
mx1.b.example.org. IN HINFO "NO ADDRESS RECORDS"

An algorithm for SMTP sender in a dual-stack environment is as follows:

(1) Lookup the MX record for the destination domain. If a CNAME record
    is returned, go to the top of step (1) with replacing the
destination domain by the query’s result. If any MX records are
    returned, go to step (2) with the query’s result (Implicit MX). If
    NO_DATA (i.e. empty answer with NOERROR(0) RCODE) is returned,
    there is no MX record but other records (e.g. SOA, NS or A etc.)
    may be found. Go to step (3). If HOST_NOT_FOUND (i.e. empty
    answer with NXDOMAIN(3) RCODE) is returned, there is no such
domain. Raise a permanent email delivery failure. Finish.

    NOTE: Some guard mechanism must required to break circular CNAME
    references.

(2) Compare each host name in MX records with the name of sending host.
    If there is a record which has the same name, drop MX records which
    have equal to or larger than preference value of the matched MX
    record (including itself). If multiple MX records remain, sort the
    MX records in ascending order based on their preference values.
    Loop over steps (3) to (9) on each host name in MX records in a
    sequence. If no MX records remain, the sending host must be the
    primary MX host. Other routing rule should be applied. Finish.

(3) If the sending MTA has IPv4 capability, lookup the A record. Keep
    the resulting address until step (5).

(4) If the sending MTA has IPv6 capability, lookup the AAAA record.

    NOTE: IPv6 addresses for hosts defined by MX records may be
    informed in additional information section of DNS querie’s result
    as well as IPv4 addresses. If there is no additional address
    information for the MX hosts, separate queries for A or AAAA
    records should be sent. There is no way to query A and AAAA
    records at once in current DNS implementation.

(5) If there is no A or AAAA record present, try the next MX record (go
    to step (3)).

    NOTE: If one or more address records are found, some MTA
    implementation may sort addresses based on the implementation’s
    preference of A or AAAA records. To encourage the transition from
    IPv4 SMTP to IPv6 SMTP, AAAA records should take precedence. But
    this type of sorting is optional.
For each of the addresses, loop over steps (7) to (9).

Try to make a TCP connection to the destination. If successful, go to step (9).

If unsuccessful and there is another available address, try the next available address. Go to step (7). If all addresses are not reachable and if a list of MX records is being traversed, try the next MX record (go to step (3)). If there is no list of MX records, or if the end of the list of MX records has been reached, raise a temporary email delivery failure. Finish.

Try an SMTP protocol negotiation according to [RFC2821] [Klensin, 2001]. If a transient failure condition reported, try the next MX record (go to step (3)). If an error condition reported, raise a permanent email delivery error, and further MX records are not tried. Finish. If successful, SMTP delivery has succeeded. Finish.

4. MX configuration in the recipient domain

4.1. Ensuring reachability for both protocol versions

If a site has dual-stack reachability, the site SHOULD configure both A and AAAA records for its MX hosts. This will help both IPv4 and IPv6 senders to reach the site efficiently.

4.2. Reachability between the primary and secondary MX

When registering MX records in a DNS database in a dual-stack environment, reachability between MX hosts must be considered carefully. Suppose all inbound email is to be gathered at the primary MX host, "mx1.example.org."

    example.org.    IN MX   1   mx1.example.org.
    IN MX   10  mx10.example.org.
    IN MX   100 mx100.example.org.

If "mx1.example.org" is an IPv6-only node, and the others are IPv4-only nodes, there is no reachability between the primary MX host and the other MX hosts. When email reaches one of the lower MX hosts, it cannot be relayed to the primary MX host based on MX preferencing mechanism.

    ; This configuration is troublesome.
    ; No secondary MX can reach mx1.example.org.
    example.org.    IN MX   1   mx1.example.org.  ; IPv6 only
    IN MX   10  mx10.example.org.  ; IPv4 only
    IN MX   100 mx100.example.org.  ; IPv4 only

The easiest possible configuration is to configure the primary MX host as a dual-stack node. By doing so, secondary MX hosts will have no
problem reaching the primary MX host.

; This configuration works well.
; The secondary MX hosts are able to relay email to the primary MX host
; without any problems.
example.org. IN MX 1 mx1.example.org. ; dual-stack
IN MX 10 mx10.example.org. ; IPv4 only
IN MX 100 mx100.example.org. ; IPv6 only

It may not be needed that the primary MX host and lower MX hosts reach
directly one another with IPv4 or IPv6 transport. For example, it is
possible to establish a routing path with UUCP or an IPv4/v6 translator.
It is also possible to drop messages into single mailbox with shared
storage using NFS or something else offered by a dual-stack server. It
is receiver site’s matter that all messages delivered to each MX hosts
must be reached to recipient’s mail drop. In such cases, dual-stack MX
host may not be listed in the MX list.

5. Operational experience

Many of the existing IPv6-ready MTA’s appear to work in the way
documented in section 3.

>From past experiments and operational experience, it is known that most
of the existing IPv4-only MTA’s will not be confused by AAAA records
that are registered for MX hostnames. No experiments were conducted
with A6 records.

There were, however, cases where IPv6-ready MTA’s were confused by
broken DNS servers. When attempting to canonify a hostname, some broken
name servers return SERVFAIL (RCODE 2), a temporary failure, on AAAA
record lookups. Upon this temporary failure, the email is queued for a
later attempt. In the interest of IPv4/v6 interoperability, these
broken DNS servers should be fixed.

6. Open issues

○ How should scoped addresses in email addresses be interpreted on
  MTA’s? As email is relayed between MTA’s, interpretation of scoped
  addresses can be different between MTA’s. Afterall, intermediate
  MTA’s may be in different scope zones than the originator. If a
  scoped IPv6 address is returned as the result of a DNS lookup, how
  should MTA’s behave?

  If scoped addresses in ‘‘route-addr’’ specifications [Crocker, 1982]
  are considered, e.g.

  <@kame.net,@[fec0::1]:itojun@itojun.org>

  it gets even trickier. Luckily, the route-addr form was obsoleted by
7. Security considerations

As mentioned in the "Open issues" section, it could be problematic if the route-addr email address format is used across multiple scope zones. MTA's would need to reject email with improper route-addr email address formats. One example of an improper route-addr format is an email from outside the site border which carries a numeric site-local address in the route-addr format.

References


Partridge, 1986.  


Crocker, 1982.  


Change history

00 -> 01  
Corrected the email address notation for source-routed emails, based on a comment from Gregory Neil Shapiro.

01 -> 02  
Change a reference to refer to RFC2822, not 822. Used "example.org", not "sample.org". These changes were based on comments from Arnt Gulbrandsen. Added an "Operational experiences" section. Clarified the case where an MX record points to a CNAME record, based on comments from Mohsen Souissi.
In some cases, IPv6-ready MTA’s are troubled by incorrect DNS server responses for AAAA queries. This change was based on comments from Gregory Neil Shapiro.

Grammar cleanups by JJ Behrens. More text on the delivery error cases.

Change title, suggested by Alain Durand.

Section on summary of IPv4 MX operation is deleted (Replaced by Introduction). Clarify on CNAME chain. Cleanups on sender’s algorithm. Suggested by Patrik Faltstrom.

Acknowledgements

This draft was written based on discussions with Japanese IPv6 users and help from the WIDE research group. Here is a (probably incomplete) list of people who contributed to the draft: Gregory Neil Shapiro, Arnt Gulbrandsen, Mohsen Souissi, and JJ Behrens.

Author’s address

Motonori NAKAMURA
Center for Information and Multimedia Studies, Kyoto University
Yoshida-nihonmatsu-cho, Sakyo, Kyoto 606-8501, JAPAN
Tel: +81-75-753-9063
Fax: +81-75-753-9056
Email: motonori@media.kyoto-u.ac.jp

Jun-ichiro itojun HAGINO
Research Laboratory, Internet Initiative Japan Inc.
Takebashi Yasuda Bldg.,
3-13 Kanda Nishiki-cho,
Chiyoda-ku,Tokyo 101-0054, JAPAN
Tel: +81-3-5259-6350
Fax: +81-3-5259-6351
Email: itojun@iiijlab.net