SMTP operational experience in mixed IPv4/IPv6 environments
draft-ietf-ngtrans-ipv6-smtp-requirement-05.txt

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. Summary of IPv4 MX operation

For reference purposes, this section outlines how email message delivery is performed in an IPv4-only environment [Partridge, 1986].
In IPv4 SMTP operation, the MX record "example.org." would be registered as follows:

```
example.org.            IN MX   1  mx1.example.org.
mx1.example.org.        IN A    192.0.2.1
mx10.example.org.       IN A    192.0.2.2
```

When an MTA wishes to deliver a message to a particular destination (e.g. "foo@example.org"), the MTA sends DNS queries in the following order:

- Lookup MX record for "example.org.".
- If an MX record is returned, lookup an A record for the right-hand side of the MX record.
- If a CNAME record is returned, try to chase the CNAME chain. Eventually an A record will be reached.

NOTE: RFC2181 [Elz, 1997] prohibits MX records from pointing to CNAME records. However, this was not prohibited in earlier RFCs. [Partridge, 1986] CNAME chasing logic is mentioned here just for backwards compatibility. Implementers may want to avoid CNAME chasing to better conform with RFC2181.

- If the MX lookup fails with NO_DATA, it means that there is no MX record, but there may be other records (e.g. "example.org."). Lookup the A record for "example.org.".
- If the MX lookup fails with HOST_NOT_FOUND, it means that there is no record at all for "example.org.". This results in a delivery failure.

2. MX records and dual stack SMTP operation

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

Similar to the way RFC’s for IPv6 DNS lookup [Thomson, 1995; Crawford, 2000] use IN class for both IPv4 and IPv6, IN MX records will be used for both IPv4 and IPv6.

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 [StJohns, 1993], however, may require special consideration when translators are used.
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.

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.

```
; 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.
mxl.a.example.org.      IN AAAA 3ffe:501::ffff:1 ; IPv6 only
b.example.org.          IN MX   1  mx1.b.example.org.
mxl.b.example.org.      IN HINFO "NO ADDRESS RECORDS"
```

(1) Lookup the MX record for the destination domain. If a CNAME record is returned, go to step (1) with the query’s result. If any MX records are returned, go to step (2) with the query’s result. If NO_DATA is returned, there is no MX record. Go to step (3). If HOST_NOT_FOUND is returned, there is no domain. Raise a permanent email delivery failure. Finish.

NOTE: the previous section contains a note about MX records that point to CNAME records.
(2) There are multiple MX records. Sort the MX records in ascending order based on their preference values, and loop over steps (3) to (8).

(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.

(5) If there is no A or AAAA record present, try the next MX record (go to step (3)). Sort the query's result based on the implementation's preference of A or AAAA records. If it is desirable to encourage the transition from IPv4 SMTP to IPv6 SMTP, AAAA records should take precedence.

(6) For each of the addresses or each part of the list of addresses, loop over steps (7) to (8). If no reachable destination is found, 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.

(7) Try to make a TCP connection to the destination. If unsuccessful, try the next available address. If successful, go to step (8).

(8) Try an SMTP protocol negotiation. If the SMTP protocol negotiation fails with TEMPFAIL (4xx), try the next MX record (go to step (3)). 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 entering 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 secondary MX hosts, it
cannot be relayed to the primary MX host.

; 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

There are many other ways to ensure that the primary MX host and the secondary MX hosts can reach one another. For example, it is possible to configure the secondary MX hosts to route email statically, i.e. without considering the DNS MX configuration. It is also possible to establish an alternate email routing path (e.g. UUCP or an IPv4/v6 translator) between the secondary MX host and the primary MX host.

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, 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 RFC2822 [Resnick, 2001].

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.


StJohns, 1993.


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.

02 -> 03
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.

03 -> 04
Grammar cleanups by JJ Behrens. More text on the delivery error
cases.

04 -> 05
Change title, suggested by Alain Durand.

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, Sakyō, 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@iijlab.net