Internet Engineering Task Force
Internet-Draft
Expires: December 16, 2003

Y. Morishita
JPRS
T. Jinmei
Toshiba

June 17, 2003

Common Misbehavior against DNS Queries for IPv6 Addresses
draft-morishita-dnsop-misbehavior-against-aaaa-00.txt

Status of this Memo

Copyright Notice

Abstract

There is some known misbehavior of DNS authoritative servers when they are queried for AAAA resource records. Such behavior can block IPv4 communication which should actually be available, cause a significant delay in name resolution, or even make a denial of service attack. This memo describes details of the known cases and discusses the effect.

1. Introduction

Many DNS clients (resolvers) that support IPv6 first search for AAAA

RRs (Resource Records) of a target host name, and then for A RRs of the same name. This fallback mechanism is based on the DNS specifications. Thus, if a DNS server which is responsible for the name is not compliant to the specifications, unpleasant results can happen. In some cases, for example, a web browser fails to connect to a web server otherwise it could. In the following sections, this memo describes some typical cases of the misbehavior, the rationale, and (bad) effects of them.

This memo shows concrete implementations and domain names that may cause problematic cases so that the behavior can be reproduced in a practical environment. The examples are for informational purposes only, and the authors do not intend accusation against any implementations or zone administrators described in this memo.

2. Network Model

In this memo, we assume a typical network model of name resolution environment using DNS. It consists of three components; stub resolvers, caching servers, and authoritative servers. A stub resolver issues a recursive query to a caching server, which then handles the entire name resolution procedure recursively. The caching server caches the result of the query as well as sends the result to the stub resolver. The authoritative servers respond to queries for names for which they have the authority, normally in a non-recursive manner.

3. Expected Behavior

Suppose that an authoritative server has an A RR but not a AAAA for a host name. Then the server should return a response to a query for a AAAA RR of the name with the RCODE being 0 (indicating no error) and with an empty answer section [1]. Such a response indicates that there is at least one RR of a different type than AAAA for the queried name, and the stub resolver can then look for A RRs.

This way, the caching server can cache the fact that the queried name does not have a AAAA RR (but may have other types of RRs), and thus can improve the response time to further queries for a AAAA RR of the name.

4. Problematic Behaviors

There are some known cases not compliant to the expected behavior. This section describes those problematic cases.

4.1 Return NXDOMAIN

This type of server returns a response with the RCODE being 3 (NXDOMAIN) to a query for a AAAA RR, indicating it does not have any RRs of any type for the queried name. In fact, such a server apparently returns NXDOMAIN to all queries except those for an A RR.

With this response, the stub resolver may immediately give up and never fall back. Even if the resolver retries with a query for an A RR, the negative response for the name has been cached in the caching server, and the caching server will simply return the negative response. As a result, the stub resolver considers this as a fatal error in name resolution.

An example of this case was found by looking for a AAAA RR of www.css.vtext.com at 66.174.3.4, although the implementation of the authoritative server seemed to change to that described in the next section.

4.2 Return NOTIMP

Other authoritative servers return a response with the RCODE being 4 (NOTIMP), indicating the servers do not support the requested type of query.

This case is less harmful than the previous one; if the stub resolver falls back to querying for an A RR, the caching server will process the query correctly and return an appropriate response.

In this case, the caching server does not cache the fact that the queried name has no AAAA RR, resulting in redundant queries for AAAA RRs in the future. The behavior will waste network bandwidth and increase the load of the authoritative server.

The current implementation of an authoritative server for css.vtext.com looks to belong to this category.

Using SERVFAIL or FORMERR would cause the same effect, though the authors have not seen such implementations yet.

4.3 Ignore Queries for AAAA

Some authoritative severs seem to ignore queries for a AAAA RR, causing a delay to fall back to a query for an A RR. This behavior may even cause a fatal timeout at the stub resolver.

This can be seen by trying to ask for a AAAA RR of "ftp-mozilla.gftp-mozilla.netscape.com," which is an alias of ftp.mozilla.org, at

205.188.139.70.

Again, these servers apparently ignore all queries except those for an A RR.

4.4 Return a Broken Response

Some other type of authoritative servers return broken responses to AAAA queries.

An example of such a response can be seen by querying for a AAAA RR of "www.gslb.mainichi.co.jp" at 210.173.172.2. This authoritative server returns a response whose RR type is AAAA, but the length of the RDATA is 4 bytes. The 4-byte data looks like the IPv4 address of the queried host name. That is, the RR in the answer section would be described like this:

```
  www.gslb.mainichi.co.jp. 600 IN AAAA 210.158.208.73
```

which is, of course, bogus (or at least meaningless).

The same behavior can be found with the name vip.alt.ihp.sony.co.jp (which is an alias of www.sony.co.jp) at 210.139.255.204.

BIND 8 caching servers transparently return the broken response (as well as cache it) to the stub resolver. BIND 9 caching servers parse the response by themselves, and send a separate response with the RCODE being 2 (SERVFAIL).

In the former case, many stub resolvers consider this as a fatal error, and do not fall back to querying for an A RR. This is the case for the BIND resolver library and (reportedly) that implemented in Internet Explorer on Windows XP SP1.

In the latter case, if the stub resolver retries the query for an A RR, it will get an appropriate response.

There are reportedly other kinds of resolver implementations that can fall back to queries for an A RR even in the first case, but the authors actually do not know of such implementations.

4.5 Make a Delegation Loop

Some authoritative servers constantly indicate a (loop) delegation for any queries except those for an A RR.

For example, such a server would return a response to a query for a AAAA RR of "www.bad.example" as follows:

```
    www.bad.example. IN NS ns.foo.bad.example.
    ns.foo.bad.example. IN A 10.0.0.1
```

Then the caching server will ask 10.0.0.1 for a AAAA RR of "www.bad.example" and see the same answer.

Caching servers interpret this as a lame delegation, and return a response with the RCODE being 2 (SERVFAIL) to the stub resolver. Furthermore, BIND 8 caching servers record the authoritative server as lame and will not use it for a certain period of time. BIND 9 caching servers relax the rule a little bit. They basically try to avoid using the lame server, but still continue to try it as a last resort.

With a BIND 8 caching server, even if the stub resolver falls back to querying for an A RR, the caching server will simply return a response with the RCODE being SERVFAIL, since all the servers are known to be "lame."

This behavior was previously found by asking for a AAAA RR of "www.united.com" at 64.95.89.4, which has recently been fixed.

5. Security Considerations

The CERT/CC pointed out that the response with NXDOMAIN described in Section 4.1 can be used for a denial of service attack [2]. The same argument applies to the cases of "broken responses" and "delegation loop" described in Section 4.4 and Section 4.5, respectively.

6. Acknowledgements

Erik Nordmark encouraged the authors to publish this document as an Internet Draft. Akira Kato and Paul Vixie reviewed a preliminary version of this draft.

Normative References

[1] Mockapetris, P., "DOMAIN NAMES - CONCEPTS AND FACILITIES", RFC 1034, November 1987.

[2] The CERT Coordination Center, "Incorrect NXDOMAIN responses from AAAA queries could cause denial-of-service conditions", March 2003, <http://www.kb.cert.org/vuls/id/714121>.

Authors' Addresses

MORISHITA Orange Yasuhiro
Research and Development Department, Japan Registry Service Co.,Ltd.
Fuundo Bldg 3F, 1-2 Kanda-Ogawamachi
Chiyoda-ku, Tokyo 101-0052
Japan

EMail: yasuhiro@jprs.co.jp

JINMEI Tatuya
Corporate Research & Development Center, Toshiba Corporation
1 Komukai Toshiba-cho, Saiwai-ku
Kawasaki-shi, Kanagawa 212-8582
Japan

EMail: jinmei@isl.rdc.toshiba.co.jp

Full Copyright Statement

Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.