

Response Differences between NSD and other DNS Servers

Jelte Jansen*, *NLnet Labs*
Wouter Wijngaards†, *NLnet Labs*

NLnet Labs document 2006-004

November 2, 2006

Abstract

This note describes observed differences in responses between NSD and other DNS server implementations. NSD 3.0.0 is compared to NSD 2.3.6, BIND 8.4.7 and BIND 9.3.2. Differences in answers to captured queries from resolvers are tallied and analyzed. No interoperability problems are found.

*jelte@nlnetlabs.nl

†wouter@nlnetlabs.nl

Contents

1	Introduction	4
2	Response differences between BIND 9.3.2 and NSD 3.0.0	5
2.1	Comparison of responses to root queries	5
2.2	Comparison of responses to NL TLD queries	5
2.3	Features	6
2.3.1	n-clrdo bit - NSD clears DO bit in response	6
2.3.2	n-clrdbit - NSD clears CD bit in response	6
2.3.3	b-class0 - CLASS0 formerr in BIND	7
2.3.4	n-tcinquery - TC bit in query is formerr for NSD	7
2.3.5	b-soattl - BIND sets SOA TTL in authority section to 0 for SOA queries	7
2.3.6	b-classany-nxdomain - BIND gives an auth answer for class ANY nxdomain	8
2.3.7	b-badquery-badanswer - BIND replies with bad answer for some bad queries	9
2.4	Functionality Differences	9
2.4.1	d-notify - different NOTIFY errors	10
2.4.2	n-update - NSD does not implement dynamic update . . .	10
2.4.3	b-mailb - BIND does not implement MAILB	10
2.4.4	d-version - BIND returns servfail on version.server queries	10
2.4.5	d-additional - Different additional section on truncated an- swers	10
2.4.6	d-refusedquery - BIND includes query section in REFUSED answers	10
2.4.7	d-hostname - BIND adds a NS record for hostname.bind .	11
2.4.8	n-ixfr-notimpl - NSD does not implement IXFR	11
2.4.9	d-formerrquery - BIND includes query section in FOR- MERR answers	11
2.4.10	d-badqueryflags - BIND includes query section in FOR- MERR answers	11
2.4.11	d-unknown-class - BIND includes query section in answers to unknown class	11
2.4.12	d-unknown-opcode - NSD returns NOTIMPL for unknown opcode	11
2.4.13	b-upwards-ref - BIND returns root delegation	12
2.4.14	b-noglue-nsquery - BIND returns no glue for NS queries .	12
2.4.15	d-noquestion - different error on no question	12
2.4.16	b-uchar - BIND returns FORMERR on strange characters	12
3	Response differences between NSD 2.3.6 and NSD 3.0.0	12
3.1	Comparison of responses in root trace	12
3.2	Comparison of responses in NL TLD trace	13
3.3	Version number - version.bind and version.server	13

CONTENTS

3.4	n-notify - notify not implemented in NSD 2	13
3.5	n-ixfr - IXFR error FORMERR in NSD 2	13
4	Response differences between BIND 8 and NSD 3.0.0	14
4.1	Comparison of responses in root trace	14
4.2	Comparison of responses in NL TLD trace	14
4.3	b8-nodata-ttlminup - BIND 8 uses minimum TTL from SOA also if bigger	15
4.4	b8-badquery-ignored - BIND 8 replies normally for some bad queries	15
4.5	b8-badedns0 - BIND 8 ignores bad EDNS0 queries	15
4.6	b8-auth-any - BIND 8 includes an authority section on queries for ANY	15
4.7	b8-ignore-tc-query - BIND 8 ignores the TC bit in queries	16

1 Introduction

The NSD name server is compared to other DNS server implementations in order to assess server interoperability. The goal is to observe differences in the answers that the name servers provide. These differences are categorized and counted.

We used BIND 8 and BIND 9 versions to compare against. Also regression tests have been run on our testlab, comparing NSD 2 versus NSD 3.

Our method uses a set of queries captured from production name servers. These queries are sent over UDP to a name server set up to serve a particular zone. Then the responses from the name server are recorded. For every query, the different answers provided by the server implementations are compared.

Unparseable answers and no answers from the servers are handled identically by the comparison software. This is not a problem because both BIND and NSD are mature and stable DNS implementations, all answers they send are parseable. Only in a very few cases, where the query is very badly formed, no answers are sent back.

The differences are found by replaying captured DNS query traces from the NL TLD and from the root zone against different name servers. The differences in the answers are then analyzed, by first performing a byte-comparison on the packets. If the packets are binary different, the contents are parsed, thus removing differences in domain name compression, and normalized (sorted, lowercase) in presentation. If the results do not match after normalization, then a list of difference categories is consulted. The difference is classified as the first category that matches. If a difference in answers does not match any category, then the process stops and the user is notified. All the differences are categorized for the traces we present.

In addition, we gratefully made use of the PROTOS DNS tool developed at the University of Oulu which they made publicly available at the protos web-page¹ and played the queries against the authoritative name servers. We fixed a packet parsing error in NSD3-prerelease and both NSD3 and BIND 9.3.2 remained running and responsive.

Additionally we used the faulty DNS query traces in the wiki-ethereal repository. These can be found in the ethereal wiki². These traces posed no problem for BIND and NSD, mostly FORMERR answers.

A previous document DIFFERENCES between BIND 8.4.4 and NSD 2.0.0 can be found in the NSD 2.x package.

In the places where differences have been found between BIND and NSD, in the authors' opinion, no interoperability problems result for resolvers.

¹<http://www.ee.oulu.fi/research/ouspg/protos/testing/c09/dns>

²<http://wiki.ethereal.com/SampleCaptures>

2 Response differences between BIND 9.3.2 and NSD 3.0.0

In this section the response differences between BIND 9.3.2 and NSD 3.0.0 are presented and analyzed. We start in Section 2.1 and Section 2.2 with presenting the difference statistics for two test traces. Then in Section 2.3 and Section 2.4 the difference categories are explained in more detail.

2.1 Comparison of responses to root queries

Comparison between NSD 3.0.0 and BIND 9.3.2 for a root trace.

<i>difference</i>	<i>packets</i>	<i>%diff</i>
d-additional (2.4.5)	455607	59.19%
n-clrdobit (2.3.1)	208389	27.07%
b-soattl (2.3.5)	101707	13.21%
n-update (2.4.2)	1858	0.24%
d-hostname (2.4.7)	1032	0.13%
d-formerrquery (2.4.9)	773	0.10%
b-class0 (2.3.3)	264	0.03%
d-refusedquery (2.4.6)	79	0.01%
d-notify (2.4.1)	18	0.00%
b-mailb (2.4.3)	7	0.00%
n-tcinquery (2.3.4)	6	0.00%
b-classany-nxdomain (2.3.6)	5	0.00%
d-badqueryflags (2.4.10)	4	0.00%
n-ixfr-notimpl (2.4.8)	3	0.00%
d-version (2.4.4)	1	0.00%
Total number of differences:	769753	100%
Number of packets the same after normalization:	1474863	
Number of packets exactly the same on the wire:	59161	
Total number of packets inspected:	2244616	

For each type of difference the number of packets in the trace that match that difference are shown. The section where that difference is analyzed is shown in parenthesis after the difference name. The percentage of differences explained by the difference category is listed. Adding up the packets that are different gives the total number of differences, or 100% of the differences.

The number of packets after normalization includes the number of packets that are the same on the wire. The total number of query packets is displayed at the bottom of the table.

2.2 Comparison of responses to NL TLD queries

Comparison between NSD 3.0.0 and BIND 9.3.2, for a trace for .nl.

2 RESPONSE DIFFERENCES BETWEEN BIND 9.3.2 AND NSD 3.0.0

<i>difference</i>	<i>packets</i>	<i>%diff</i>
d-unknown-opcode (2.4.12)	2541	26.44%
b-badquery-badanswer (2.3.7)	1817	18.91%
n-clrdobit (2.3.1)	1495	15.56%
b-soattl (2.3.5)	1120	11.65%
n-update (2.4.2)	990	10.30%
d-badqueryflags (2.4.10)	847	8.81%
d-hostname (2.4.7)	531	5.52%
d-notify (2.4.1)	98	1.02%
b-upwards-ref (2.4.13)	78	0.81%
n-clrcdbit (2.3.2)	63	0.66%
d-version (2.4.4)	22	0.23%
b-noglue-nsquery (2.4.14)	8	0.08%
b8-badedns0 (4.5)	1	0.01%
Total number of differences:	9611	100%
Number of packets the same after normalization:	90389	
Number of packets exactly the same on the wire:	52336	
Total number of packets inspected:	100000	

2.3 Features

In this section we enumerate a number of differences between BIND 9.3.2 and NSD 3.0.0 that cannot be immediately explained as design choices. These features could be seen as bugs in software or protocol specs, except that they do not lead to interoperability problems.

2.3.1 n-clrdobit - NSD clears DO bit in response

NSD clears the DO bit in answers to queries with the DO bit. BIND copies the DO bit to the answer.

Analysis: In RFC4035[1] the DO bit is not specified for answers. In the examples section of that RFC the DO bit is shown for signed dig responses, although this could refer to the query or the answer. NSD clears the DO bit for all answers, a decision based on speed: the EDNS record sent back by NSD is precompiled and not modified during answer processing.

2.3.2 n-clrcdbit - NSD clears CD bit in response

NSD clears the CD bit in answers to queries with the CD bit. BIND copies the CD bit to the answer.

Analysis: RFC 4035[1] asserts that the CD bit must be cleared for authoritative answers. The CD bit should be copied into the answer by recursive servers. BIND copies the CD bit for some formerr queries.

2 RESPONSE DIFFERENCES BETWEEN BIND 9.3.2 AND NSD 3.0.0

2.3.3 b-class0 - CLASS0 formerr in BIND

For CLASS0, you can get either FORMERR, from BIND or REFUSED, from NSD.

Analysis: Difference in interpretation of the RFCs, a CLASS value of 0 is interpreted as a syntax error by BIND but as another valid class (that is not served) by NSD. Resolvers are unaffected for CLASS IN.

2.3.4 n-tcinquery - TC bit in query is formerr for NSD

NSD returns FORMERR if tc bit is set in query.

Analysis: Queries cannot be longer than 512 octets, since the DNS header is short and the query DNS name has a maximum length of 255 octets. Thus TC (TrunCation) cannot happen. Only one question per query packet is answered by NSD, this is a design decision.

Some update, ixfr request, notify, gss-tsig TKEY sequence queries could theoretically carry longer data in the query from the client. In practice this does not happen, as 255 octet uncompressed names are not used. If this were to happen, the client could attempt a TCP connection immediately instead of setting a TC bit, or use EDNS0 to send longer packets.

In this NSD is more strict in validation than BIND.

2.3.5 b-soattl - BIND sets SOA TTL in authority section to 0 for SOA queries

This happens when asking for the SOA for a domain that is not served.

Query:

```
;; ->>HEADER<<- opcode: QUERY, rcode: NOERROR, id: 0
;; flags: rd ; QUERY: 1, ANSWER: 0, AUTHORITY: 0, ADDITIONAL: 0
;; QUESTION SECTION:
;; foo.bar.      IN      SOA
```

Answer from BIND 9.3.2:

```
;; ->>HEADER<<- opcode: QUERY, rcode: NXDOMAIN, id: 6097
;; flags: qr aa rd ; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 0
;; QUESTION SECTION:
;; foo.bar.      IN      SOA

;; ANSWER SECTION:

;; AUTHORITY SECTION:
.      0      IN      SOA      A.ROOT-SERVERS.NET. NSTLD.VERISIGN-GRS.COM. (
2006072801 1800 900 604800 86400)

;; ADDITIONAL SECTION:

;; Query time: 10 msec
;; SERVER: 127.0.0.1
;; WHEN: Wed Aug 23 13:52:36 2006
;; MSG SIZE rcvd: 100
```

2 RESPONSE DIFFERENCES BETWEEN BIND 9.3.2 AND NSD 3.0.0

Answer from NSD 3:

```
;; ->>HEADER<<- opcode: QUERY, rcode: NXDOMAIN, id: 26095
;; flags: qr aa rd ; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 0
;; QUESTION SECTION:
;; foo.bar.      IN      SOA

;; ANSWER SECTION:

;; AUTHORITY SECTION:
.      86400  IN      SOA      a.root-servers.net. nstld.verisign-grs.com. (
2006072801 1800 900 604800 86400)

;; ADDITIONAL SECTION:

;; Query time: 60 msec
;; SERVER: 127.0.0.1
;; WHEN: Wed Aug 23 13:53:30 2006
;; MSG SIZE rcvd: 100
```

Analysis: BIND conforms to internet-draft draft-andrews-dnsexth-soa-discovery which has at the moment of code development not (yet) been published as RFC. NSD conforms to the RFCs.

2.3.6 b-classany-nxdomain - BIND gives an auth answer for class ANY nx-domain

A difference in behaviour for CLASS=ANY queries. For existing domains both BIND and NSD reply with AA bit cleared. For not existing domains (nxdomain) NSD replies with AA bit cleared. BIND replies with AA bit on and includes a SOA (CLASS=IN) for the zone, as for an authoritative nxdomain.

Query:

```
;; ->>HEADER<<- opcode: QUERY, rcode: NOERROR, id: 13328
;; flags: ; QUERY: 1, ANSWER: 0, AUTHORITY: 0, ADDITIONAL: 0
;; QUESTION SECTION:
;; nslabs.ru0.  ANY      MX
```

Answer from BIND 9.3.2:

```
;; ->>HEADER<<- opcode: QUERY, rcode: NXDOMAIN, id: 13328
;; flags: qr aa ; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 0
;; QUESTION SECTION:
;; nslabs.ru0.  ANY      MX

;; ANSWER SECTION:

;; AUTHORITY SECTION:
.      86400  IN      SOA      a.root-servers.net. nstld.verisign-grs.com. (
2006072801 1800 900 604800 86400)

;; ADDITIONAL SECTION:
```


2 RESPONSE DIFFERENCES BETWEEN BIND 9.3.2 AND NSD 3.0.0

```
;; Query time: 0 msec
;; WHEN: Wed Aug 23 13:58:51 2006
;; MSG SIZE rcvd: 103
```

Answer from NSD 3:

```
;; ->>HEADER<<- opcode: QUERY, rcode: NXDOMAIN, id: 13328
;; flags: qr ; QUERY: 1, ANSWER: 0, AUTHORITY: 0, ADDITIONAL: 0
;; QUESTION SECTION:
;; nslabs.ru. ANY MX

;; ANSWER SECTION:

;; AUTHORITY SECTION:

;; ADDITIONAL SECTION:

;; Query time: 0 msec
;; WHEN: Wed Aug 23 13:58:51 2006
;; MSG SIZE rcvd: 28
```

Analysis: Feature of BIND where it answers authoritatively for CLASS ANY nxdomain queries.

2.3.7 b-badquery-badanswer - BIND replies with bad answer for some bad queries

BIND replies with an answer packet that cannot be parsed, or does not answer at all. NSD always generates an answer, with the appropriate RCODE (mostly NOTIMPL and FORMERR, but also NXDOMAIN to NOTIFY queries). All these queries are malformed in some way.

A (very simple) example of a query without an answer is a query packet of 18 zero bytes. For some queries no answer only happens when BIND is presented with a trace of queries, not for a single query.

Analysis: BIND includes (part of) the unparseable question into the answer, or some internal state of BIND is affected by earlier queries.

NSD manages to answer the malformed query. Note that NSD does not answer queries that are too short, or that have the QR bit set. NSD tries to be as liberal in what it accepts as possible.

2.4 Functionality Differences

The next group of differences are due to the fact that NSD does not implement some functionality that is requested by resolvers. This is a design choice and should not cause resolver problems at all, since responses to those requests are within protocol specs.

2.4.1 d-notify - different NOTIFY errors

BIND and NSD give different errors for notify queries. The servers are started without any configuration for access control on notify. For notify messages aimed at a zone that is served, BIND 9.3.2 returns a NOERROR answer, and NSD 3 returns NOTAUTH. For notify messages on a zone that is not served (in-addr.arpa.) BIND 9.3.2 returns NOTAUTH and NSD 3 returns NXDOMAIN.

Analysis: Default configuration differs between the two packages. NSD is more strict. Error codes are different, the tools that send notifies are not affected.

2.4.2 n-update - NSD does not implement dynamic update

For UPDATE, you can get either REFUSED/NXRRSET/other RCODE from BIND 9.3.2 or NOTIMPL from nsd3.

Analysis: NSD does not implement dynamic update.

2.4.3 b-mailb - BIND does not implement MAILB

For MAILB, you can get either NOTIMPL(BIND 9) or NOERROR/NXDOMAIN(NSD 3).

Analysis: BIND does not implement queries for the MAILB type. NSD treats it as one of the RRTYPEs. MAILB is obsoleted by RFCs, the MX type is used to transfer mail information now.

2.4.4 d-version - BIND returns servfail on version.server queries

NSD returns version.server query, BIND returns servfail.

Analysis: Both NSD and BIND return version.bind queries of the chaos class. These queries differ in the version number they return, of course. BIND does not return version.server queries. This is a design decision on the part of NSD to return version.server queries with the same answer.

2.4.5 d-additional - Different additional section on truncated answers

NSD and BIND return different additional sections on truncated answers to queries from the root. These answers are 480+ bytes long.

Analysis: Not all the A and AAAA data fits into the additional section of the answer. BIND includes different names than NSD does, and BIND is observed to sometimes include one more AAAA record, less A records in the additional section. Resolvers should be unaffected.

2.4.6 d-refusedquery - BIND includes query section in REFUSED answers

BIND includes the query sent for REFUSED answers. NSD replies with only the DNS header section.

2 RESPONSE DIFFERENCES BETWEEN BIND 9.3.2 AND NSD 3.0.0

Analysis: The resolver must inspect the query ID. The error code provides sufficient information. Sending the header makes NSD replies smaller and thus more resilient to DoS attacks.

2.4.7 d-hostname - BIND adds a NS record for hostname.bind

BIND includes an additional RR in the authority section of the reply:

```
hostname.bind. 0 CH NS hostname.bind.
```

Analysis: The RR seems useless. NSD does not include it.

2.4.8 n-ixfr-notimpl - NSD does not implement IXFR

To queries for IXFR BIND responds with a valid answer (the latest SOA) and NSD responds with NOTIMPL error.

Analysis: NSD 3.0.0 does not implement IXFR. It returns NOTIMPL by design.

2.4.9 d-formerrquery - BIND includes query section in FORMERR answers

BIND includes the query sent for FORMERR answers. NSD replies with only the DNS header section. For some queries, NSD includes an EDNS record in the reply if there was a recognizable EDNS record in the query.

Analysis: The resolver must inspect the query ID. The error code provides sufficient information. Sending the header makes NSD replies smaller and thus more resilient to DoS attacks.

2.4.10 d-badqueryflags - BIND includes query section in FORMERR answers

BIND includes the query section in reply to unparseable queries. NSD does not.

Analysis: Same as d-formerrquery (2.4.9), but the implementation of the comparison software could not parse the query either, thus a separate label.

2.4.11 d-unknown-class - BIND includes query section in answers to unknown class

For queries with an unknown class in the query, BIND includes the query section in the answer. NSD does not.

Analysis: Same as d-formerrquery (2.4.9), but for a different error.

2.4.12 d-unknown-opcode - NSD returns NOTIMPL for unknown opcode

For queries that are bad packets, with malformed RRs, with an unknown opcode, BIND returns a FORMERR, but NSD gives up after checking the opcode and returns NOTIMPL. NSD copies the flags from the query, and turns on the QR (query response) bit, BIND zeroes some of the flags.

3 RESPONSE DIFFERENCES BETWEEN NSD 2.3.6 AND NSD 3.0.0

Analysis: NOTIMPL is appropriate since NSD does not implement whatever functionality is being looked for.

2.4.13 b-upwards-ref - BIND returns root delegation

For queries to a domain that is not served, which can only have arrived at this server due to a lame delegation, BIND returns a root delegation. NSD returns SERVFAIL.

Analysis: By design, NSD does not know the root-servers. NSD is unable to reply as the zone is not configured, hence the SERVFAIL. This is also discussed in the REQUIREMENTS document for NSD.

2.4.14 b-noglue-nsquery - BIND returns no glue for NS queries

For queries for the NS records of the zone, BIND does not include glue for the NS records. NSD includes glue for the NS servers that lie within the zone.

Analysis: The glue saves a followup query.

2.4.15 d-noquestion - different error on no question

For queries without a question section the error code differs. NSD considers it a FORMERR. BIND returns REFUSED.

Analysis: Error code not specified for this corner case. No problems for resolvers.

2.4.16 b-uchar - BIND returns FORMERR on strange characters

BIND returns FORMERR on strange characters in the query, such as 0x00, 0xff, 0xe4, 0x20, 0x40 and so on.

Analysis: NSD does not give a formerr on these queries, it processes them. NSD normalizes names to lower case. Otherwise leaves them untouched. BIND preserves case in answers. Choice made in REQUIREMENTS for NSD, also see RFC1035[2] 2.3.3.

3 Response differences between NSD 2.3.6 and NSD 3.0.0

The differences between NSD 2.3.6 and NSD 3.0.0 are listed below. All are due to version number changes and new features in NSD 3.

3.1 Comparison of responses in root trace

Differences between NSD 2.3.6 and NSD 3.0.0 for a root trace. Note that apart from the 26 packets that are different, all responses are binary the same on the wire between the two versions of NSD.

3 RESPONSE DIFFERENCES BETWEEN NSD 2.3.6 AND NSD 3.0.0

<i>difference</i>	<i>packets</i>	<i>%diff</i>
n-notify (3.4)	19	73.08%
n-ixfr (3.5)	3	11.54%
version.bind (3.3)	3	11.54%
version.server (3.3)	1	3.85%
Total number of differences:	26	100%
Number of packets the same after normalization:	2244590	
Number of packets exactly the same on the wire:	2244590	
Total number of packets inspected:	2244616	

3.2 Comparison of responses in NL TLD trace

Differences between NSD 2.3.6 and NSD 3.0.0 for a nl. trace. Note that apart from the 311 packets that are different, all responses are binary the same on the wire between the two versions of NSD.

<i>difference</i>	<i>packets</i>	<i>%diff</i>
n-notify (3.4)	289	92.93%
version.bind (3.3)	22	7.07%
Total number of differences:	311	100%
Number of packets the same after normalization:	99689	
Number of packets exactly the same on the wire:	99689	
Total number of packets inspected:	100000	

3.3 Version number - version.bind and version.server

To queries for version.bind and version.server the different implementations return a different version number, as they should.

Analysis: Expected. Correct version numbers are returned.

3.4 n-notify - notify not implemented in NSD 2

Notifications are handled differently. NSD 2 returns NOTIMPL error code, while NSD 3 returns NOTAUTH or NXDOMAIN error codes.

Analysis: Default config denies all notify queries for NSD 3. These answers are correct for non-existing and not authorized domains.

3.5 n-ixfr - IXFR error FORMERR in NSD 2

To IXFR query questions different error codes are given. The NSD 2 gives FORMERR (due to the RR in the authority section). NSD 3 returns NOTIMPL.

Analysis: Neither version of NSD implements IXFR. It is more appropriate to return the NOTIMPL error code in that case. Bugfix in NSD.

4 Response differences between BIND 8 and NSD 3.0.0

In this section the response differences between BIND 8.4.7 and NSD 3.0.0 are categorized and analyzed.

4.1 Comparison of responses in root trace

The differences between BIND 8.4.7 and NSD 3.0.0 when presented with queries for the root zone are below.

<i>difference</i>	<i>packets</i>	<i>%diff</i>
n-clrddbitt (2.3.2)	516372	84.39%
d-hostname (2.4.7)	53431	8.73%
d-additional (2.4.5)	32526	5.32%
b8-nodata-ttlminup (4.3)	4611	0.75%
n-update (2.4.2)	1856	0.30%
d-version (2.4.4)	1033	0.17%
b8-auth-any (4.6)	519	0.08%
b8-badedns0 (4.5)	492	0.08%
d-unknown-class (2.4.11)	482	0.08%
b-badquery-badanswer (2.3.7)	451	0.07%
b-class0 (2.3.3)	97	0.02%
d-notify (2.4.1)	18	0.00%
b8-ignore-tc-query (4.7)	6	0.00%
b8-badquery-ignored (4.4)	4	0.00%
n-ixfr-notimpl (2.4.8)	3	0.00%
b-soattl (2.3.5)	1	0.00%
Total number of differences:	611902	100%
Number of packets the same after normalization:	1632714	
Number of packets exactly the same on the wire:	2299	
Total number of packets inspected:	2244616	

4.2 Comparison of responses in NL TLD trace

The differences between BIND 8.4.7 and NSD 3.0.0 when presented with queries for the .nl zone are below.

4 RESPONSE DIFFERENCES BETWEEN BIND 8 AND NSD 3.0.0

<i>difference</i>	<i>packets</i>	<i>%diff</i>
n-clrcdbit (2.3.2)	2857	33.53%
d-unknown-opcode (2.4.12)	2692	31.59%
n-update (2.4.2)	1283	15.06%
d-badqueryflags (2.4.10)	841	9.87%
d-hostname (2.4.7)	531	6.23%
d-notify (2.4.1)	293	3.44%
d-version (2.4.4)	22	0.26%
b-badquery-badanswer (2.3.7)	1	0.01%
b8-badedns0 (4.5)	1	0.01%
Total number of differences:	8521	100%
Number of packets the same after normalization:	91479	
Number of packets exactly the same on the wire:	90837	
Total number of packets inspected:	100000	

4.3 b8-nodata-ttlminup - BIND 8 uses minimum TTL from SOA also if bigger

For NXDOMAIN queries in root-servers.net BIND 8 uses the minimum TTL from the SOA as the TTL of the included SOA RR. However, this minimum TTL is larger than the original TTL of the SOA, both NSD 2.3.6, NSD 3 and BIND 9 use the smaller of those two values as the TTL of the included SOA.

Analysis: Bug in BIND 8 solved in BIND 9.

4.4 b8-badquery-ignored - BIND 8 replies normally for some bad queries

BIND8 manages to reply for malformed queries. NSD replies with FORMERR.

Analysis: The query is bad, formerr is needed. Fixed in BIND9.

4.5 b8-badedns0 - BIND 8 ignores bad EDNS0 queries

BIND 8 ignores queries with bad EDNS0 section. It answers the query. NSD replies with FORMERR.

Analysis: BIND8 is more liberal in accepting broken EDNS0 records. NSD is not. Changed in BIND 9.

4.6 b8-auth-any - BIND 8 includes an authority section on queries for ANY .

BIND8 includes an authority section on queries for class ANY . BIND9 and NSD return an empty authority section.

Analysis: Fixed in BIND9.

4.7 b8-ignore-tc-query - BIND 8 ignores the TC bit in queries

BIND responds to queries that have the TC bit set. NSD gives FORMERR.

Analysis: This is like the n-tcinquery (2.3.4), except where BIND9 returns NXDOMAIN, BIND8 returns the query with qr bit set. This is fixed in BIND9. NSD is less liberal in accepting queries, it returns form error on queries with the TC bit set.

References

- [1] R. Arends, R. Austein, M. Larson, D. Massey, and S. Rose. *Protocol Modifications for the DNS Security Extensions*. RFC 4035 (Proposed Standard), March 2005. <http://www.ietf.org/rfc/rfc4035.txt>.
- [2] P.V. Mockapetris. *Domain names - implementation and specification*. RFC 1035 (Standard), November 1987. <http://www.ietf.org/rfc/rfc1035.txt>, (Updated by RFCs 1101, 1183, 1348, 1876, 1982, 1995, 1996, 2065, 2136, 2181, 2137, 2308, 2535, 2845, 3425, 3658, 4033, 4034, 4035).