DNS COMPLIANCE

Fred Baker
Internet Systems Consortium

Background - 2014

ISC was in the process of adding DNS COOKIE (RFC 7873) to BIND and we wanted to see how many servers would mishandle DNS COOKIE options and in which ways as they would be sent with every query unlike other EDNS options that are only occasionally sent.

If we were going to measure how many servers would mishandle DNS COOKIE options we may as well measure how servers mishandle all EDNS extension mechanisms and track that over time.

Initially we used a experimental EDNS option while DNS COOKIE was finalised. This was introduced in BIND 9.10.0 with a configure option to turn on sending sending the EDNS option on unix based machines and it was on by default in the Windows builds.

When DNS COOKIE was allocated a code point we updated the code to use that in BIND 9.10.3.

DNS COOKIE support is on by default for all builds as of BIND 9.11.0.

https://ednscomp.isc.org/

Test your own servers
https://ednscomp.isc.org/ednscomp

draft-ietf-dnsop-no-response-issue

We published our results at http://ednscomp.isc.org and wrote a tool to allow anyone to test their own servers at https://ednscomp.isc.org/ednscomp.

We have also been working on a RFC to report on this issue which can be found by searching for draft-ietf-dnsop-no-response-issue.

Testing Method

- A series of queries for the SOA/DNSKEY RR-set at the zone's apex which tested specific aspects of EDNS behaviour.
- The responses were then examined to see if they matched the expected behaviour of a server that implements EDNS correctly.

We tested both individual extension mechanism and extension mechanisms in combination (e.g. EDNS version 1 with a EDNS option).

This talk doesn't show the combinational response errors.

Type Testing

https://ednscomp.isc.org/compliance/tld-typereport.txt

. @2001:7fd::1 (k.root-servers.net.): all ok
. @199.7.83.42 (l.root-servers.net.): URI=notimp
. @2001:500:9f::42 (l.root-servers.net.): all ok
. @202.12.27.33 (m.root-servers.net.): all ok
. @2001:dc3::35 (m.root-servers.net.): all ok

We also have been testing TLD servers for how they handle allocated and unknown type codes.

This is a small snippet of the report. In this case it show l.root-servers.net returning NOTIMP to a URI test query.

On the whole TLD servers are well behaved.

Other DNS testing.

https://ednscomp.isc.org/compliance/tld-fullreport.txt

. @2001:503:ba3e::2:30 (a.root-servers.net.): dns=ok aa=ok ad=ok cd=ok ra=ok rd=ok tc=ok zflag=ok opcode=ok opcodeflg=reset type666=ok tcp=ok edns=ok edns1=ok edns@512=ok ednsopt=ok edns1opt=ok do=ok edns1do=ok ednsflags=ok optlist=ok ednsnsid=ok ednscookie=ok ednsexpire=ok ednssubnet=ok edns1nsid=ok edns1cookie=ok edns1expire=ok edns1subnet=ok signed=ok,yes ednstcp=ok

. @192.228.79.201 (b.root-servers.net.): dns=ok aa=ok ad=ok cd=ok ra=ok rd=ok tc=ok zflag=ok opcode=ok opcodeflg=rd,cd type666=ok tcp=ok edns=ok edns1=ok edns@512=ok ednsopt=ok edns1opt=ok do=ok edns1do=ok ednsflags=ok optlist=ok,nsid ednsnsid=ok,nsid ednscookie=ok ednsexpire=ok ednssubnet=ok edns1nsid=ok edns1cookie=ok edns1expire=ok edns1subnet=ok signed=ok,yes ednstcp=ok

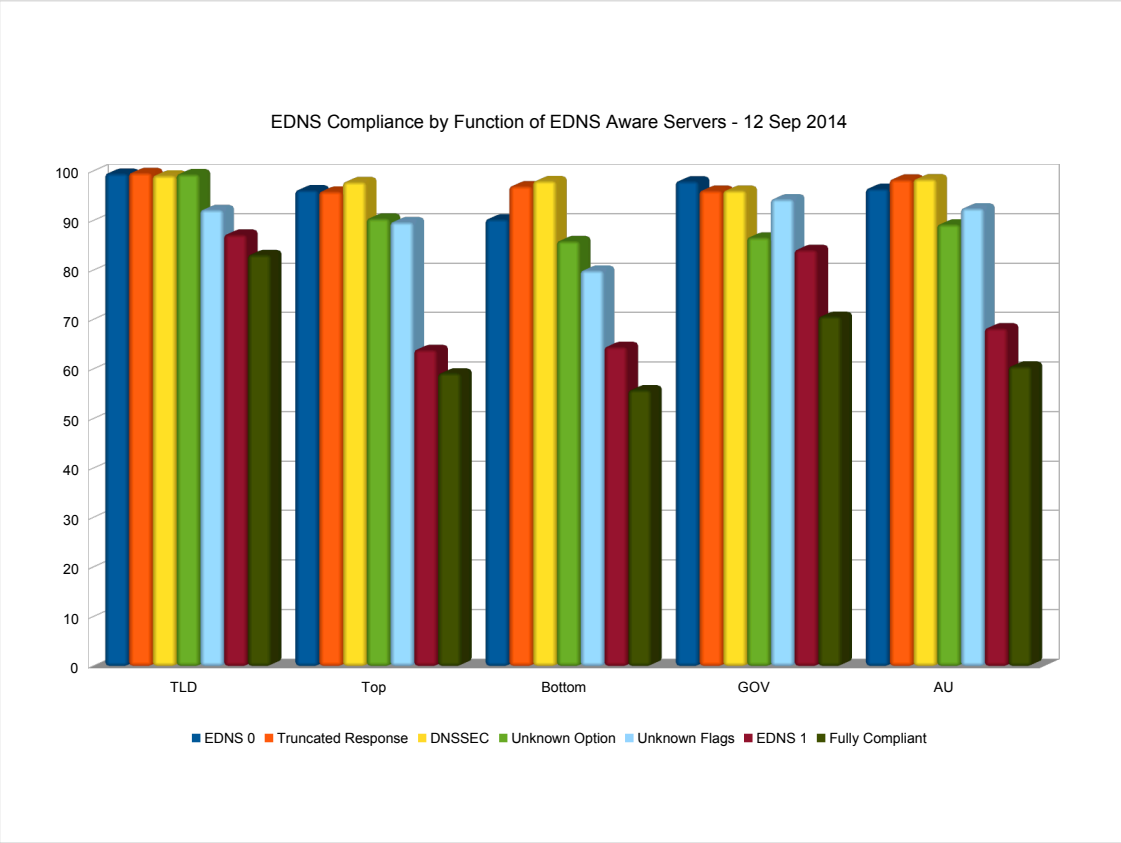
We also have been testing servers to see how they respond to having other attributes set in the query.

Some like zflag and opcode because we expect that these will be used at some point in the future and having a understanding of which servers mishandle them will be useful.

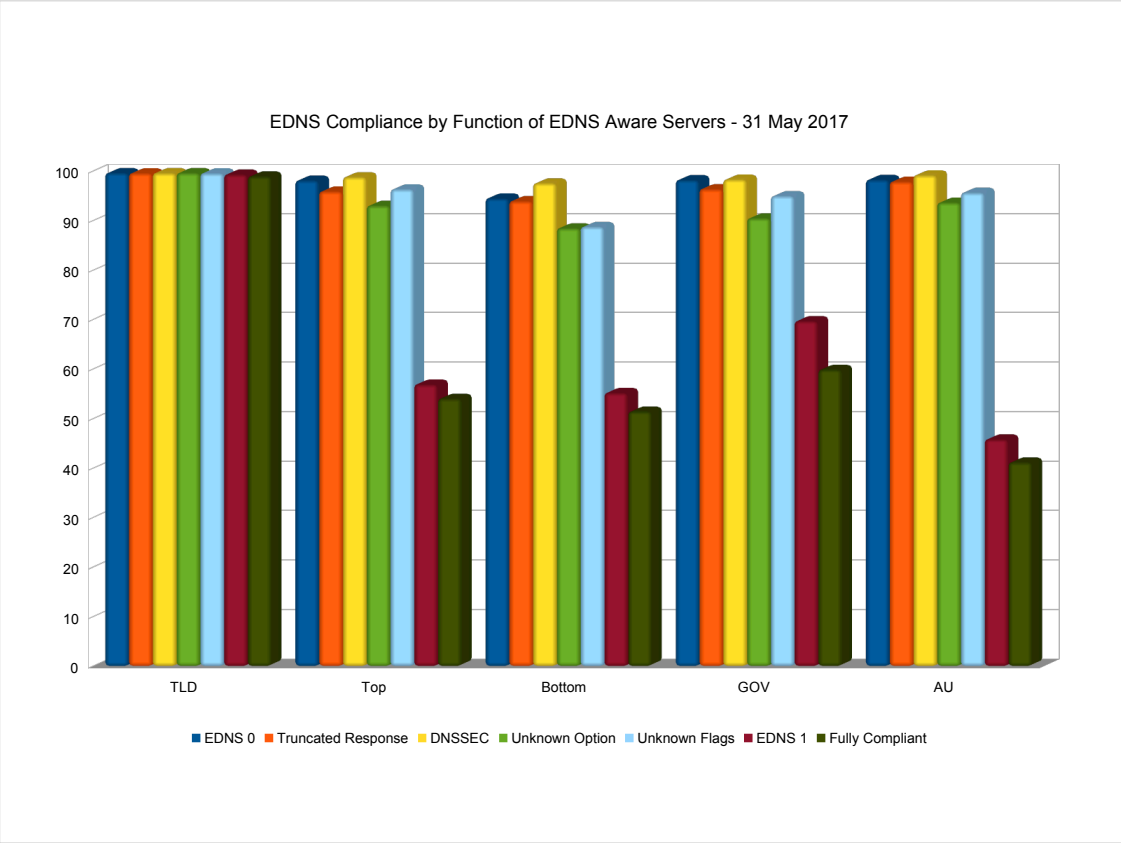
Others like AD and CD to determine which servers are currently broken. There are servers that drop queries with these flags set.

Aims of talk

- To show the current state of EDNS compliance
- To show the impact of what will happen when different EDNS extension mechanism are used without taking proactive steps to fix the current issues



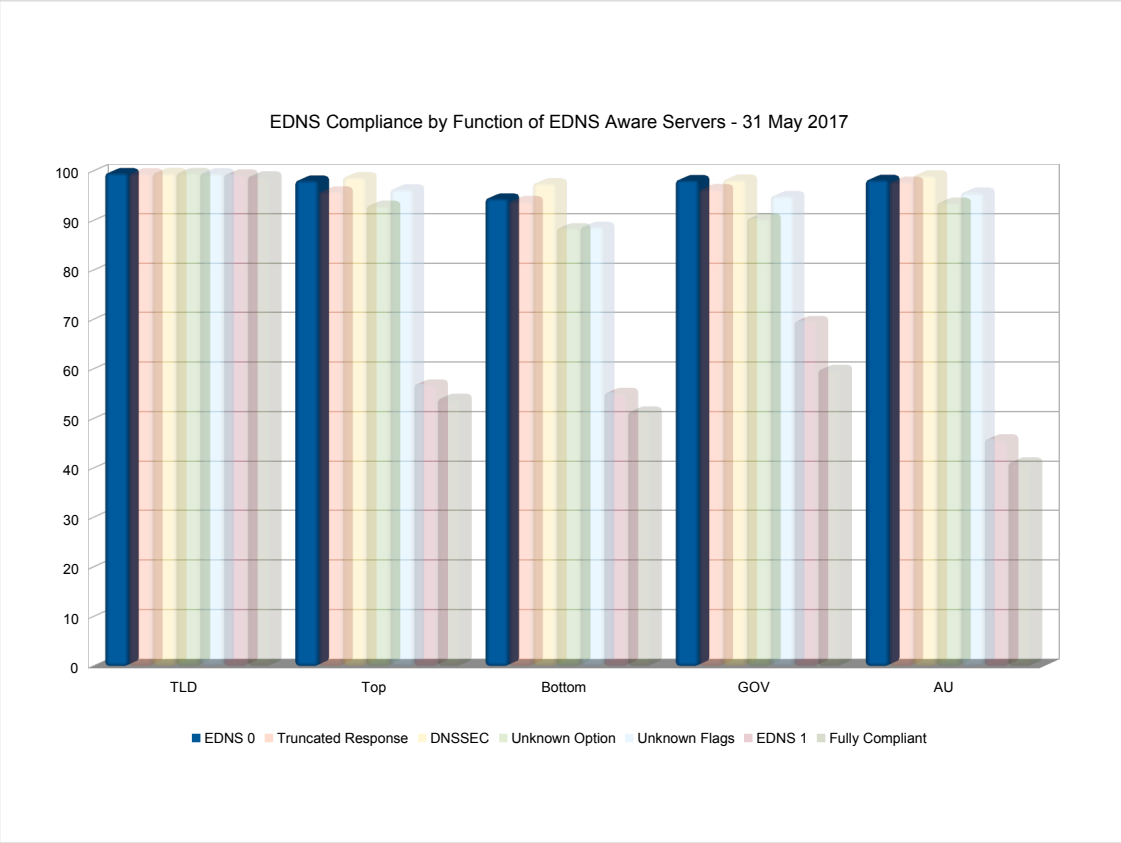
When we started we found was that DNSSEC was reasonably well supported but that unknown EDNS options and unknown EDNS flags were not well supported, and that unknown EDNS versions were very poorly supported. This was despite RFC specifying how to handle unknown EDNS options and unknown EDNS versions.



Today the behaviour is better for some of the sample sets and worse for others.

The 5 grouping are the name servers listed in the root zone. The name servers for the Top and Bottom 100 names in the Alexa Top 1 million names. The name servers for .GOV and .AU names in the Alexa Top 1M.

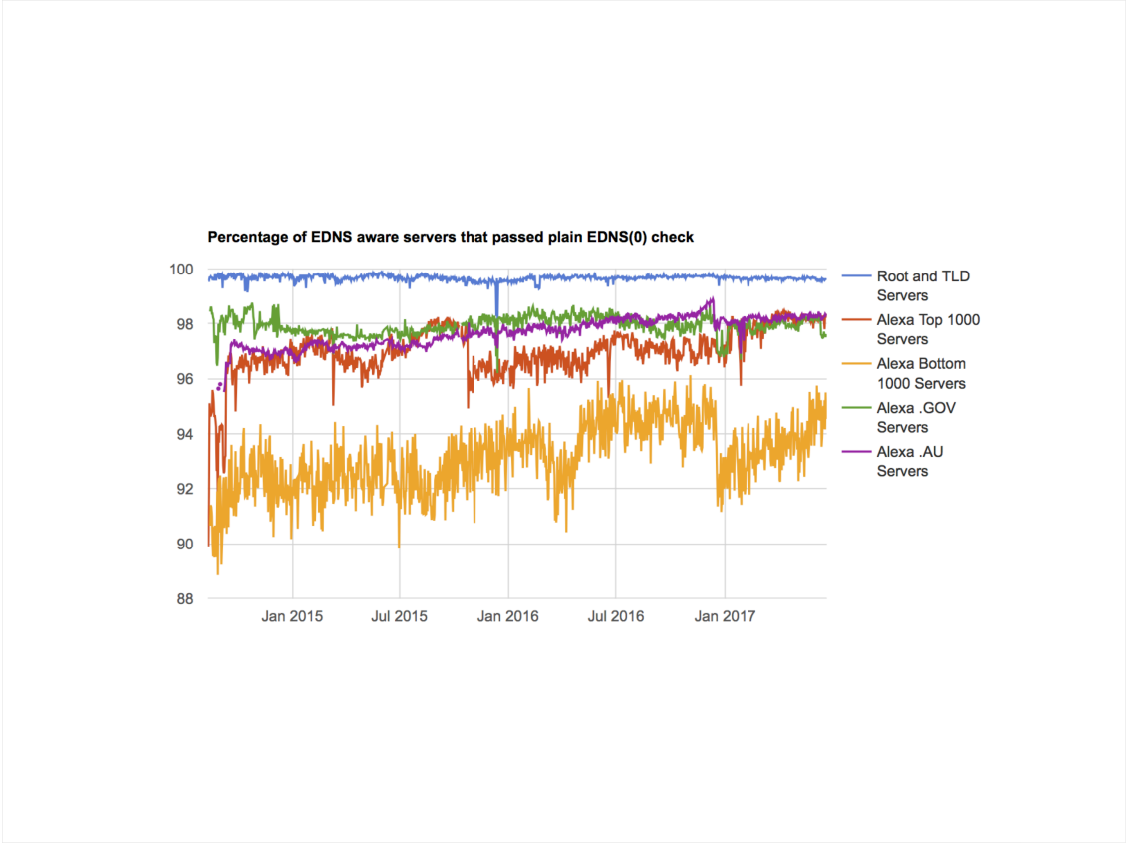
These were chosen to see if there was a difference between either end. .GOV names were chose because US Federal .GOV zones are supposed to be DNSSEC signed. .AU names were chosen because the researcher is Australian.



The blue columns are EDNS version 0 queries with no EDNS options or flags present for the SOA record at the zone's apex.

The gaps here are servers that do not respond to EDNS queries with EDNS responses unless is a DO=1 is set in the request or a NSID EDNS option is present in the request.

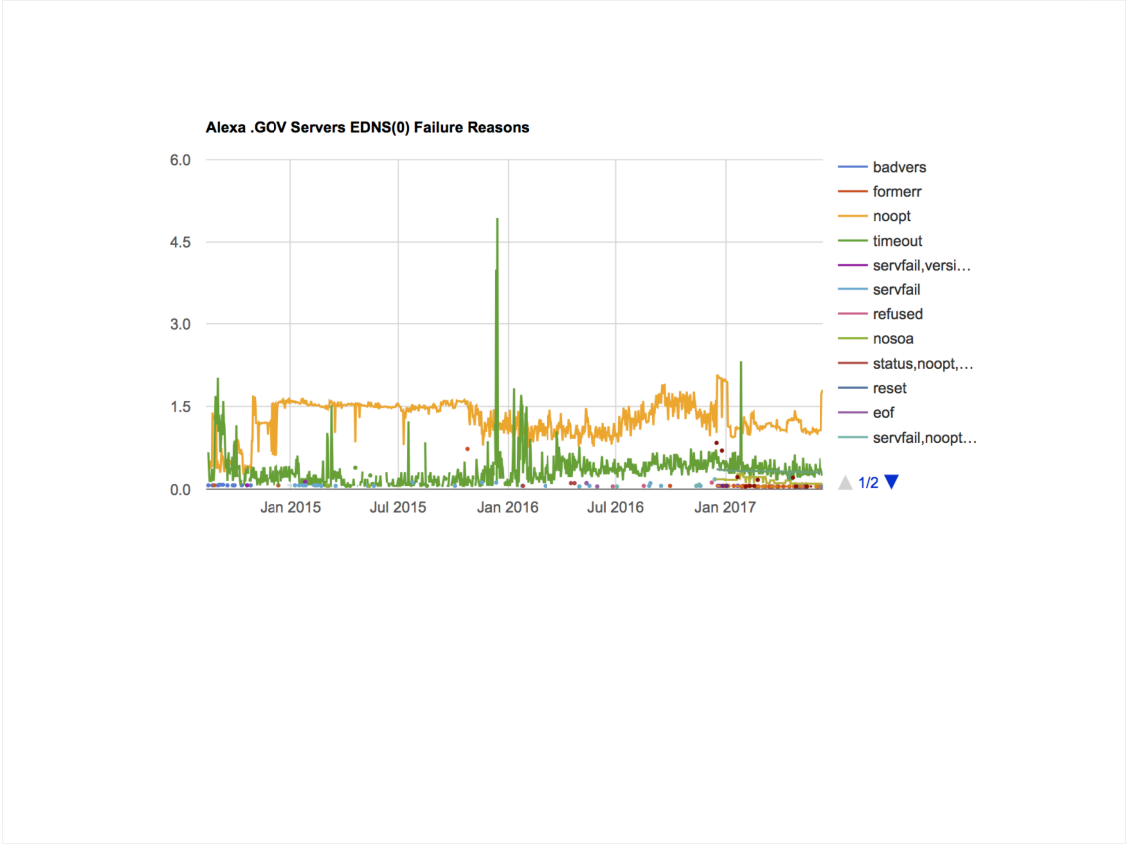
This is unspecified EDNS behaviour but is mostly benign as most clients no longer issue queries like this and this behaviour has no impact on DNSSEC validation.



EDNS(0) with no extensions over time.

The servers for bottom 1000 of the Alex top 1 million have a high level of content change which results in a noisy measurement.

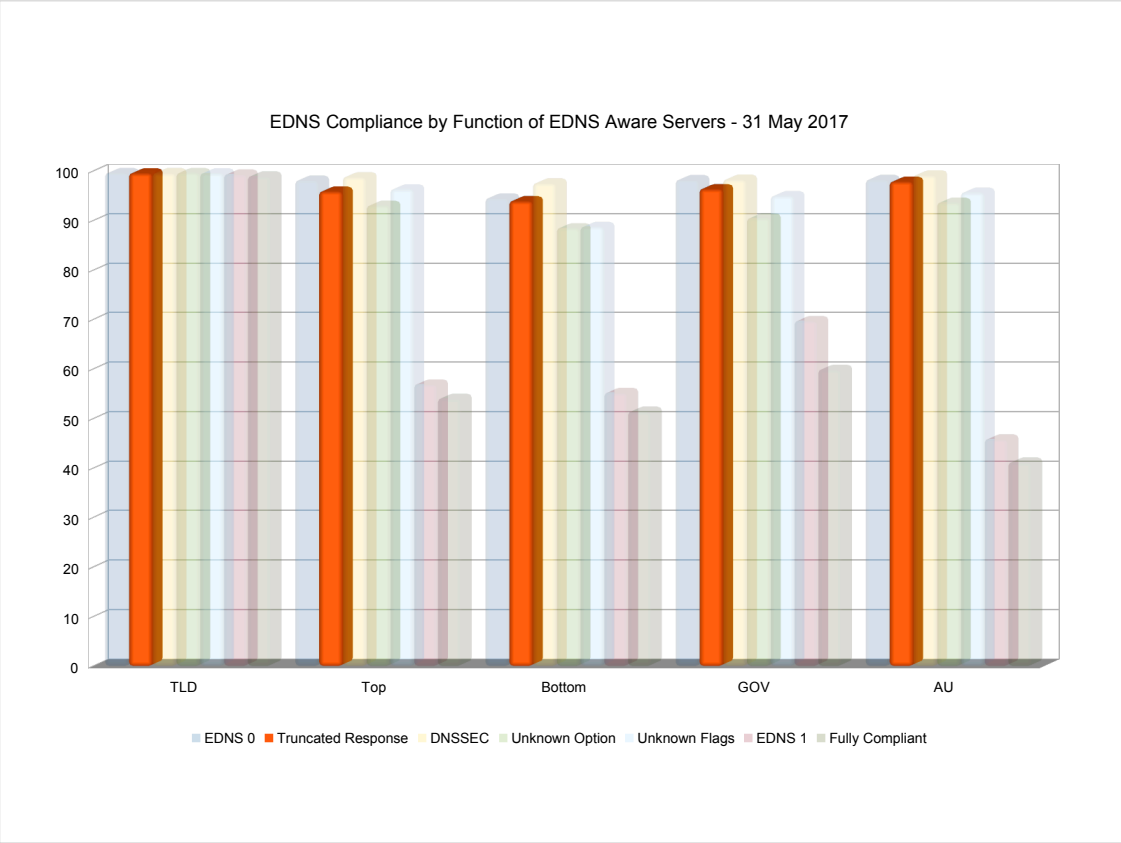
The step jump in October 2015 was when we started measuring which servers supported NSID, EXPIRE, ECS and DNS COOKIE options. This exposed servers that are EDNS aware but were not answering the existing EDNS compliance tests with a EDNS response.



This slide show EDNS Version 0 mishandling. All the servers that respond here nominally support EDNS as they gave a EDNS response to one or more of the test queries.

The yellow line shows servers that only respond to EDNS queries if DO=1, NSID or ECS option is present in the query.

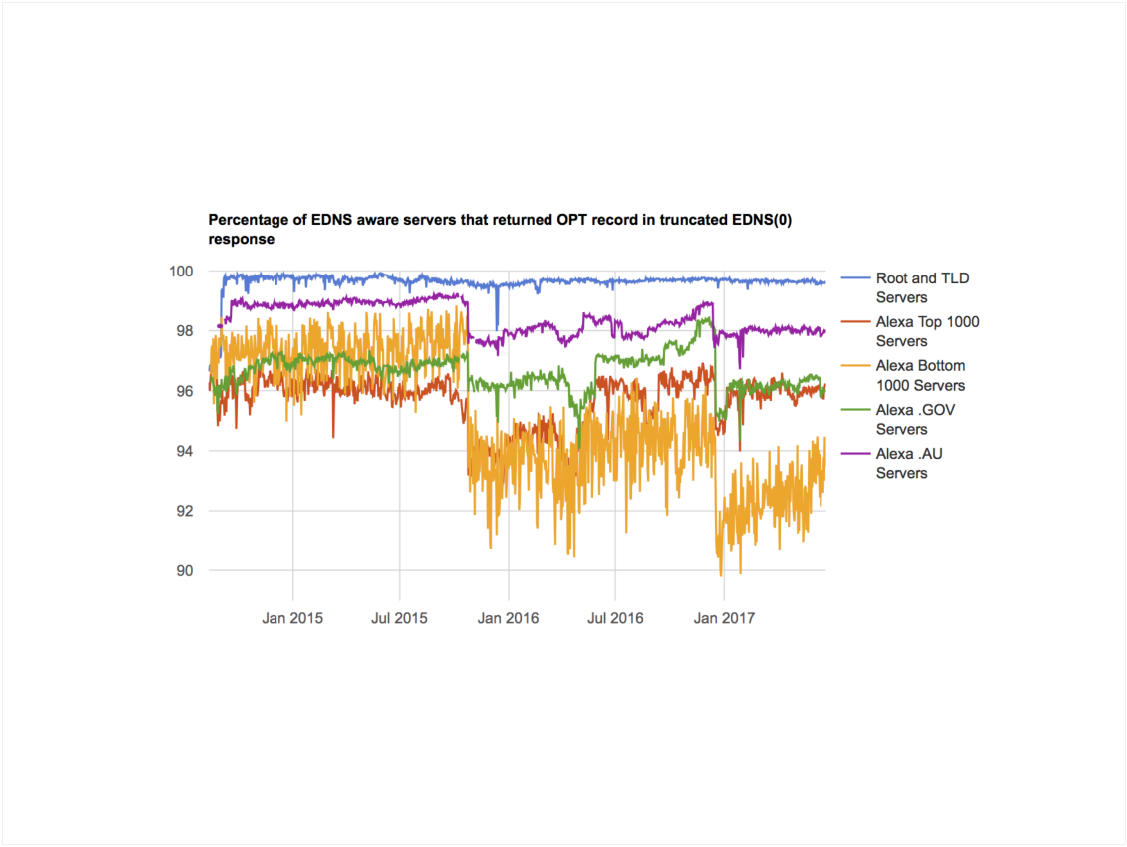
The green line show typical packet losses. Only servers that responded to at least one of the test queries are counted.

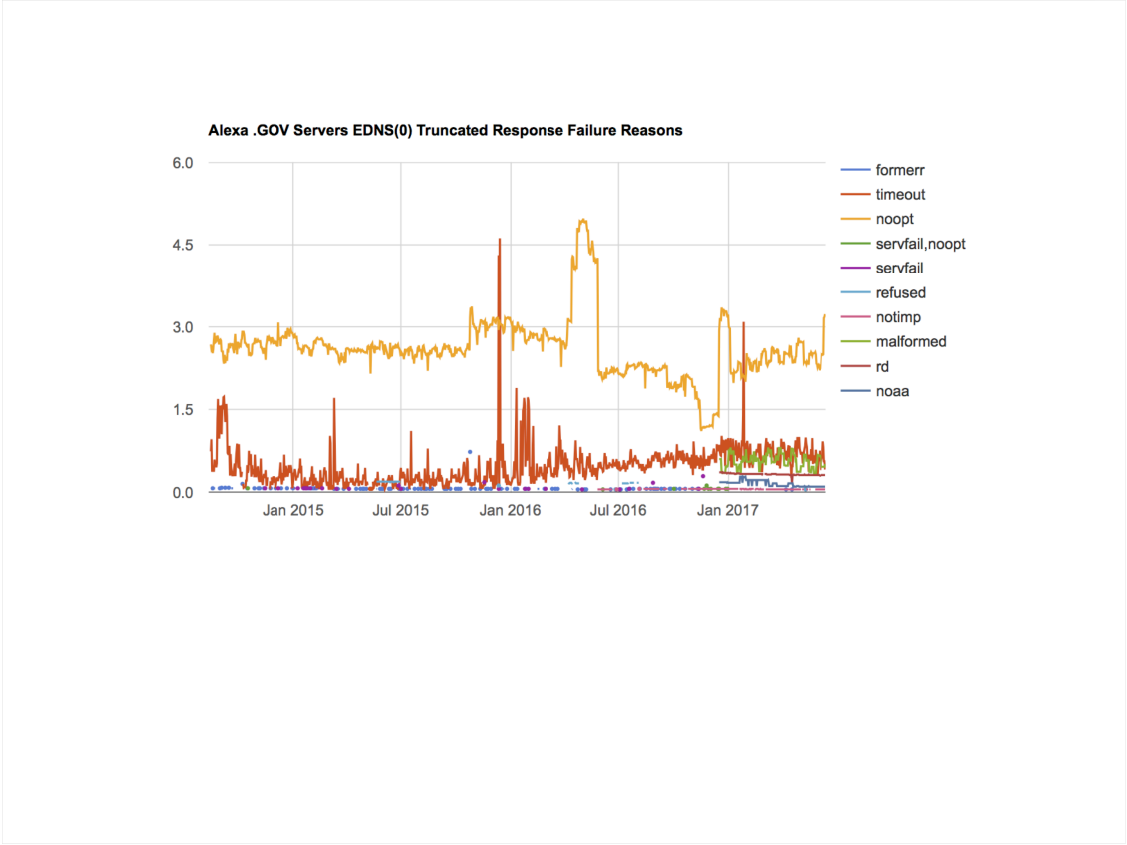


The orange columns show DNSKEY query responses to a EDNS query a EDNS UDP buffer size set to 512 bytes in a attempt to trigger a truncated UDP response from the server.

The response should have a response code of NOERROR and if EDNS is supported include a OPT record. If the zone is signed there is a high probability that the response will be truncated.

The test reports the mishandling of unsupported / unknown query types, and if the zone is signed, the mishandling of truncated responses. As there is no way to force a truncated response, the levels of misbehaviour are under reported.

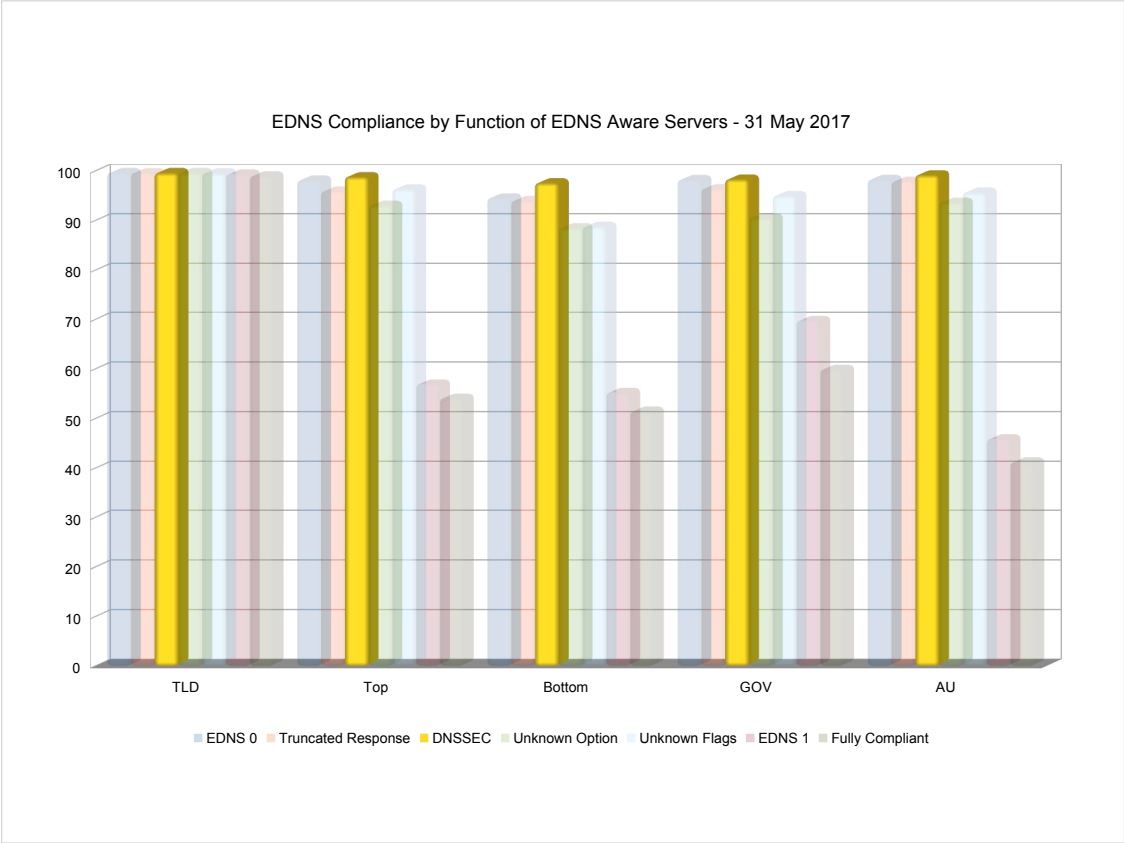

Percentage of EDNS aware servers that returned OPT record in truncated EDNS(0) response
100
98
96
94
92
90
Jan 2015
Jul 2015
Jan 2016
Jul 2016
Jan 2017
Root and TLD Servers
Alexa Top 1000 Servers
Alexa Bottom 1000 Servers
Alexa .GOV Servers
Alexa .AU Servers



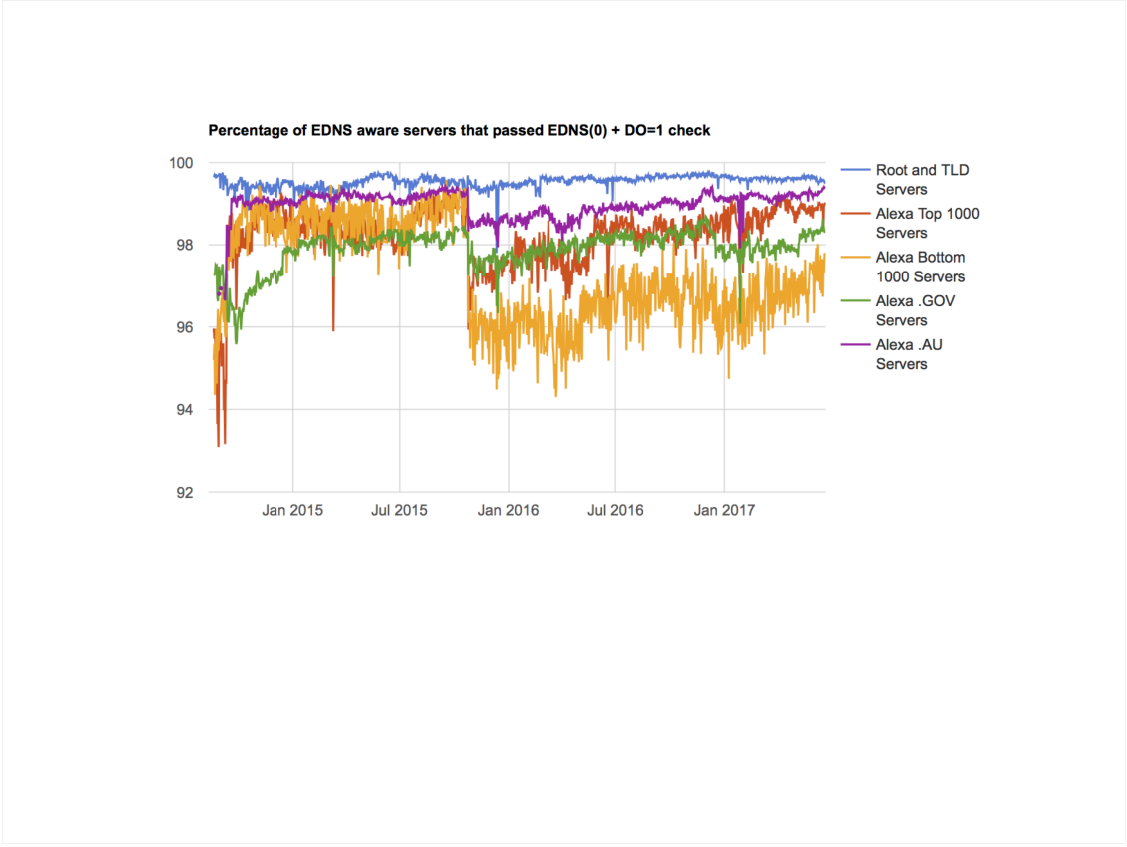
This slide shows how servers mis-generate truncated responses. The error rates will be under reported as the test does not always generate a truncated response.

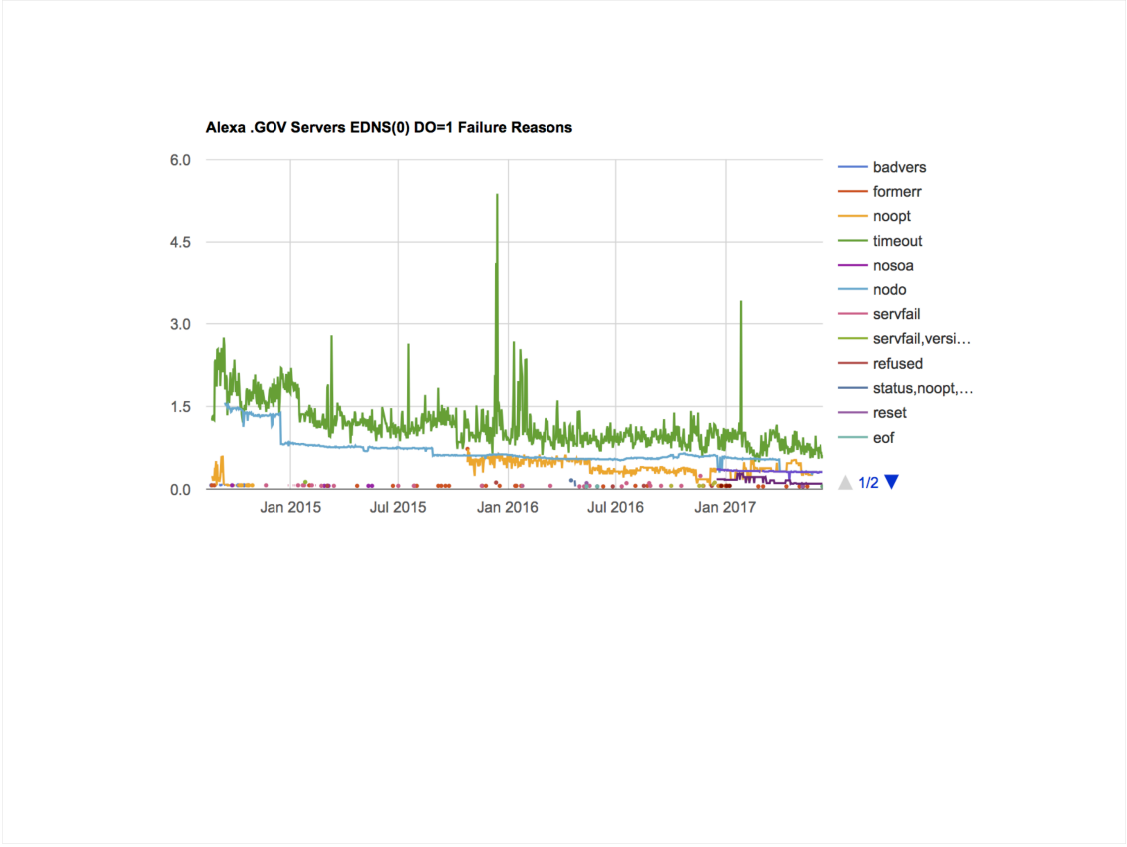
The yellow line shows the number of responses without a OPT record present. It is about 1% higher than other data sets. This risks resolvers misclassifying a server as not supporting EDNS when it gets back a truncated response. Knowing whether a server supports EDNS or not helps in determining whether a lack of response is due to packet loss or not.

This graph also shows malformed responses (the green line) and NOTIMP responses to the DNSKEY query. All those NOTIMP responses should be NOERROR no-data. The later is a negative response which can be cached.



The yellow columns show EDNS aware servers that do not mishandle DO=1 (DNSSEC) queries.


Percentage of EDNS aware servers that passed EDNS(0) + DO=1 check
100
98
96
94
92
Jan 2015
Jul 2015
Jan 2016
Jul 2016
Jan 2017
Root and TLD Servers
Alexa Top 1000 Servers
Alexa Bottom 1000 Servers
Alexa .GOV Servers
Alexa .AU Servers

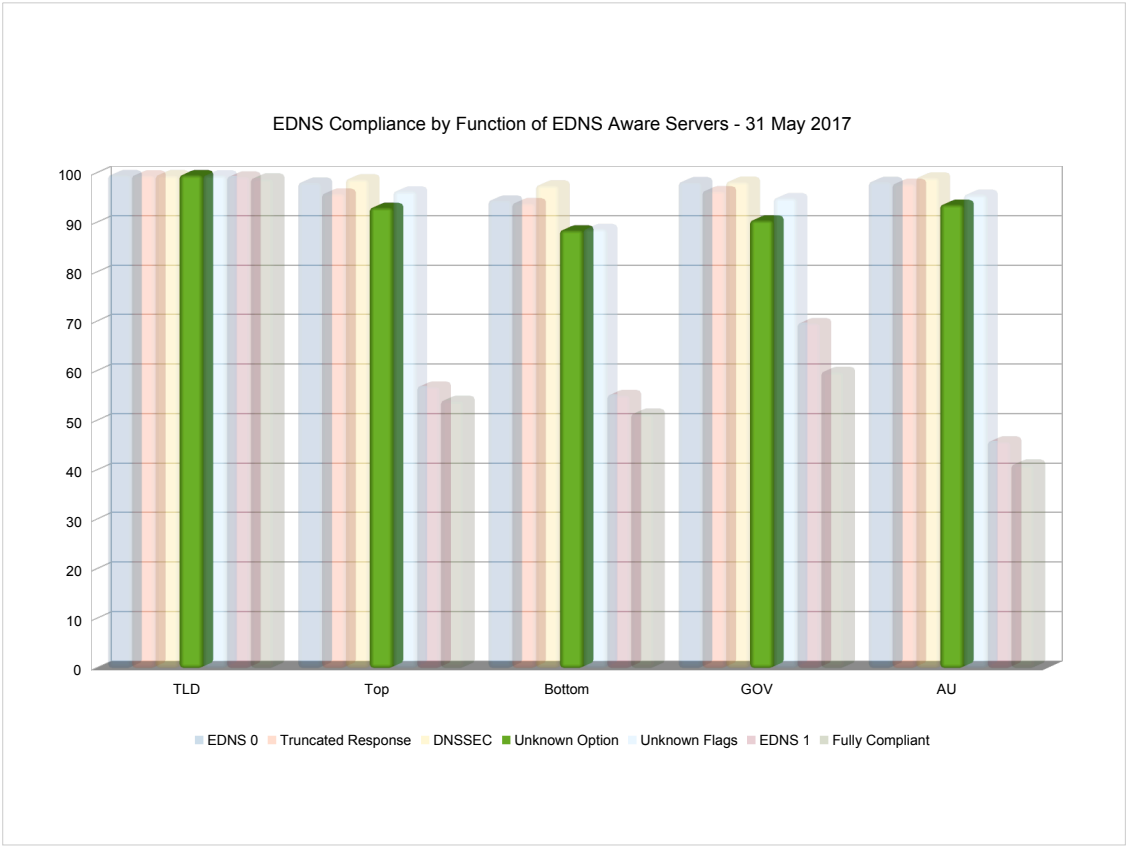


This is a break down of how EDNS DO=1 (DNSSEC) queries are mishandled.

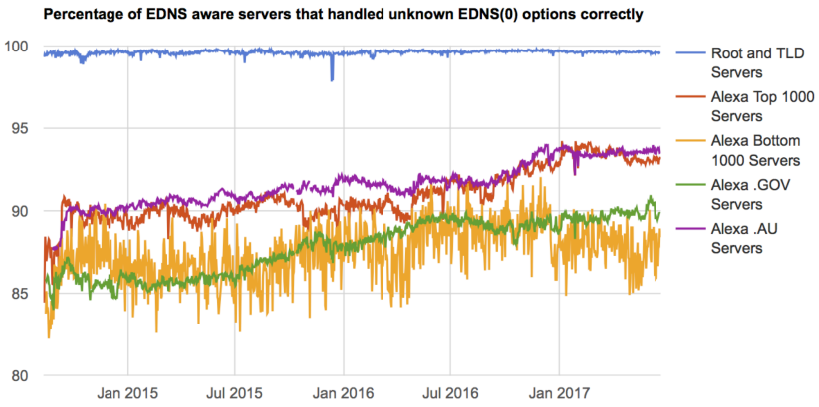
The blue line shows DNSSEC aware servers that do not set DO=1 in the response despite returning RRSIG records which indicate that they theoretically support DNSSEC.

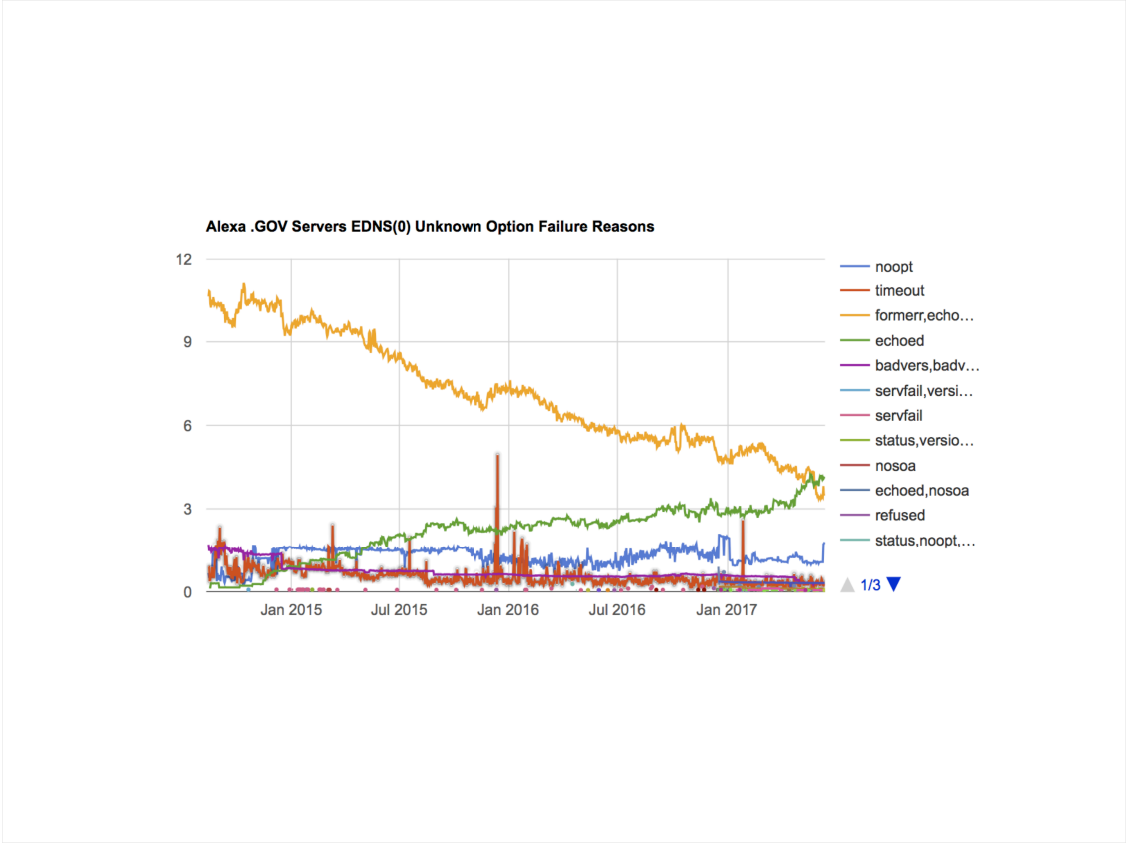
The yellow line is servers which don't return a EDNS response to DO=1 queries despite returning a EDNS response to other types of queries. Usually this a query with a EDNS NSID option present.

Packet loss is also slightly higher than non DO=1 queries. This will be due to equipment that drops fragmented responses or servers that fail to do PMTUD properly.



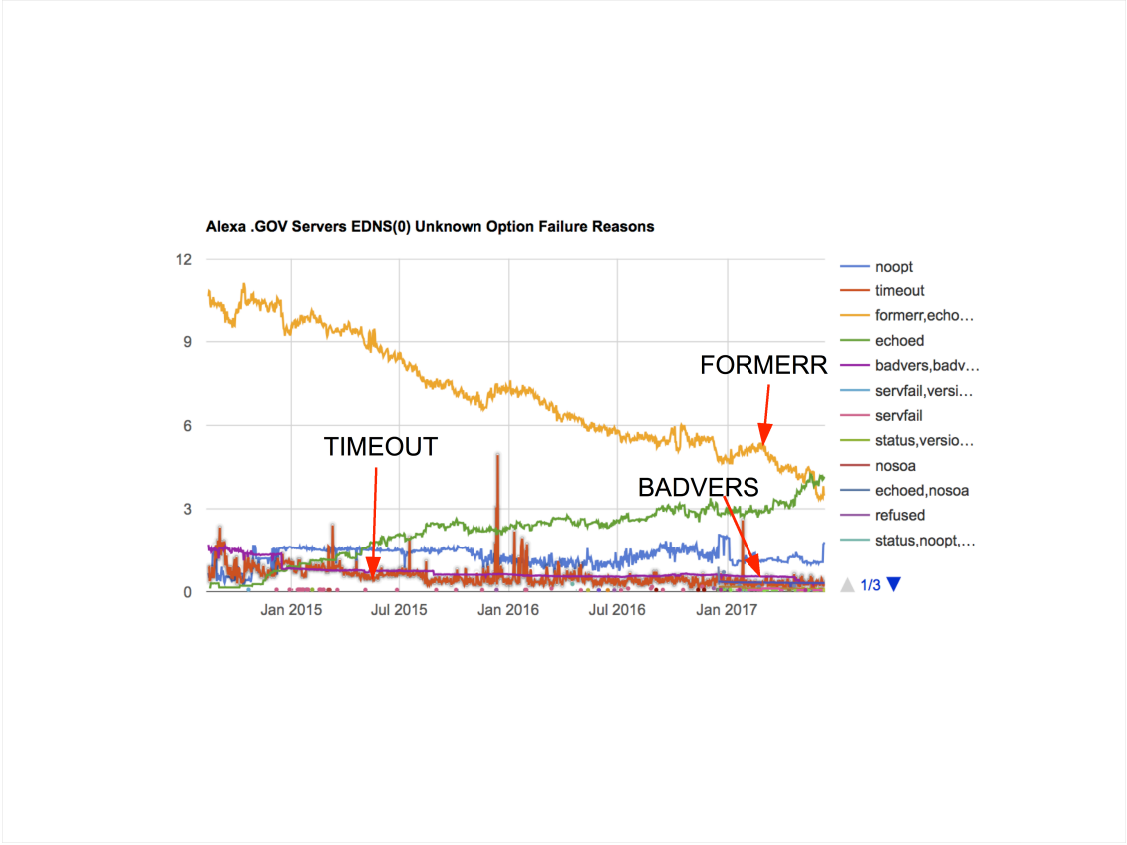
The green columns are servers which handle unknown EDNS options correctly. Unknown EDNS options are supposed to be ignored by EDNS aware servers.


Percentage of EDNS aware servers that handled unknown EDNS(0) options correctly
100
95
90
85
80
Jan 2015
Jul 2015
Jan 2016
Jul 2016
Jan 2017
Root and TLD Servers
Alexa Top 1000 Servers
Alexa Bottom 1000 Servers
Alexa .GOV Servers
Alexa .AU Servers



This is a breakdown of how unknown EDNS options are mishandled. This graph is taken from .GOV Alexa top 1 Million servers.

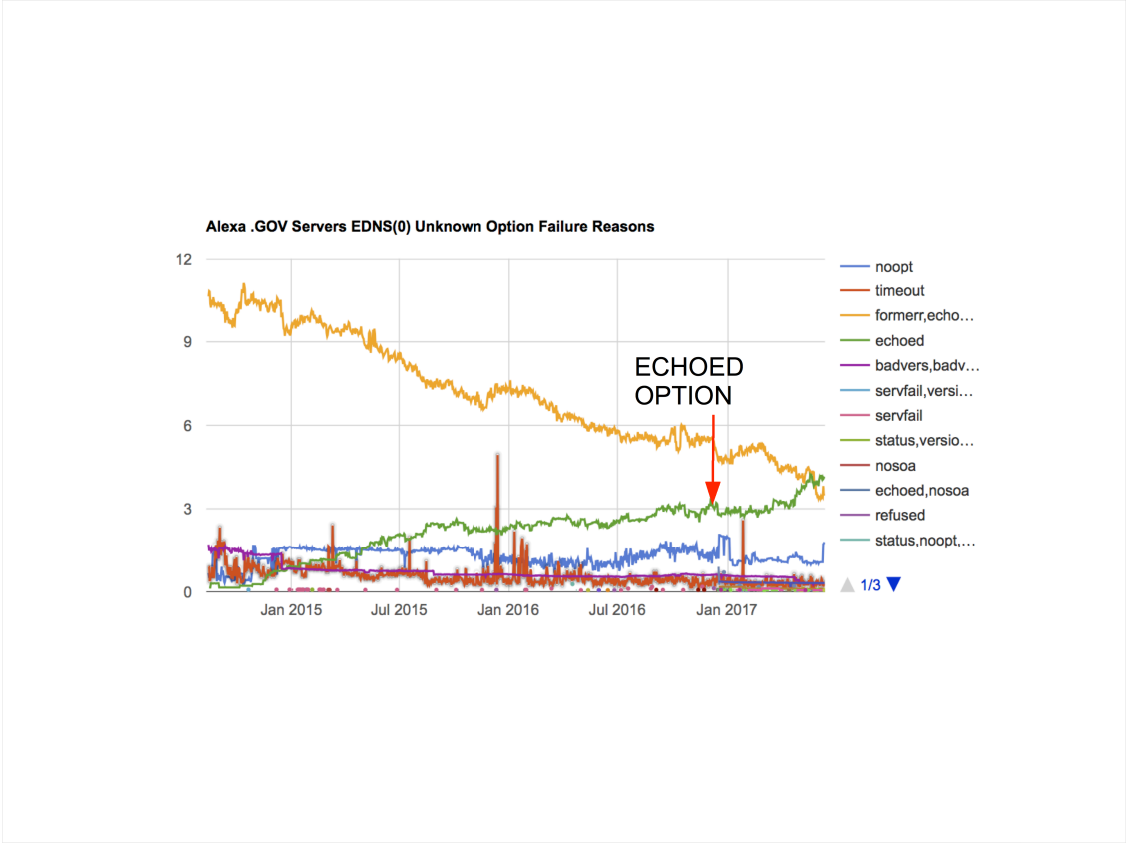
Most of the mishandling is improving with the exception of unknown EDNS options being echoed back to the client.



The three marked lines indicate servers that cause DNS resolution failures with BIND today when validating if they also serve a signed zone.

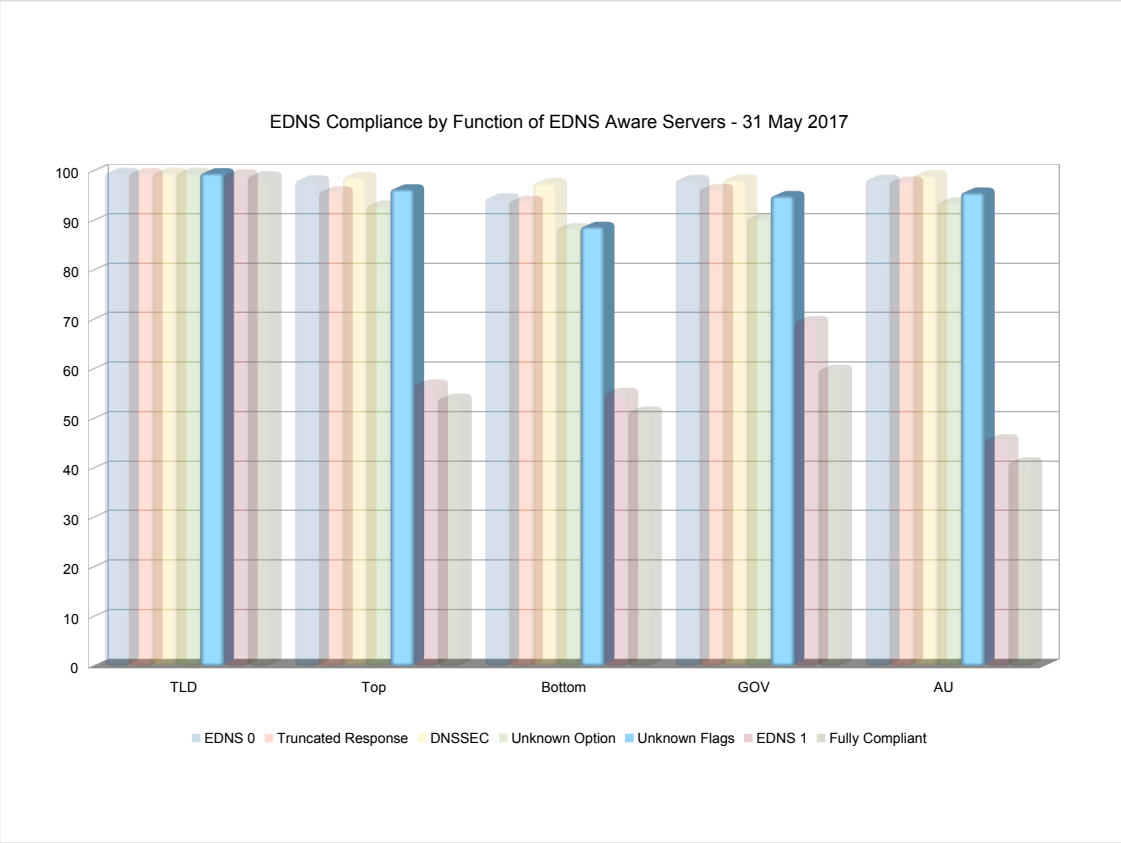
BIND treats FORMERR and BADVERS responses as a indication that the server does not support EDNS and retries the query using plain DNS which is incompatible with getting a DNSSEC response.

Similarly BIND works around servers that do not respond to EDNS queries by sending plain DNS queries. Again this results in DNSSEC validation failures.

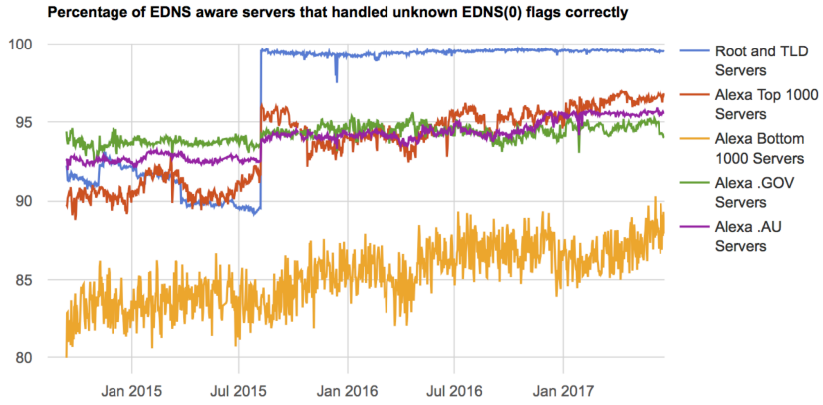


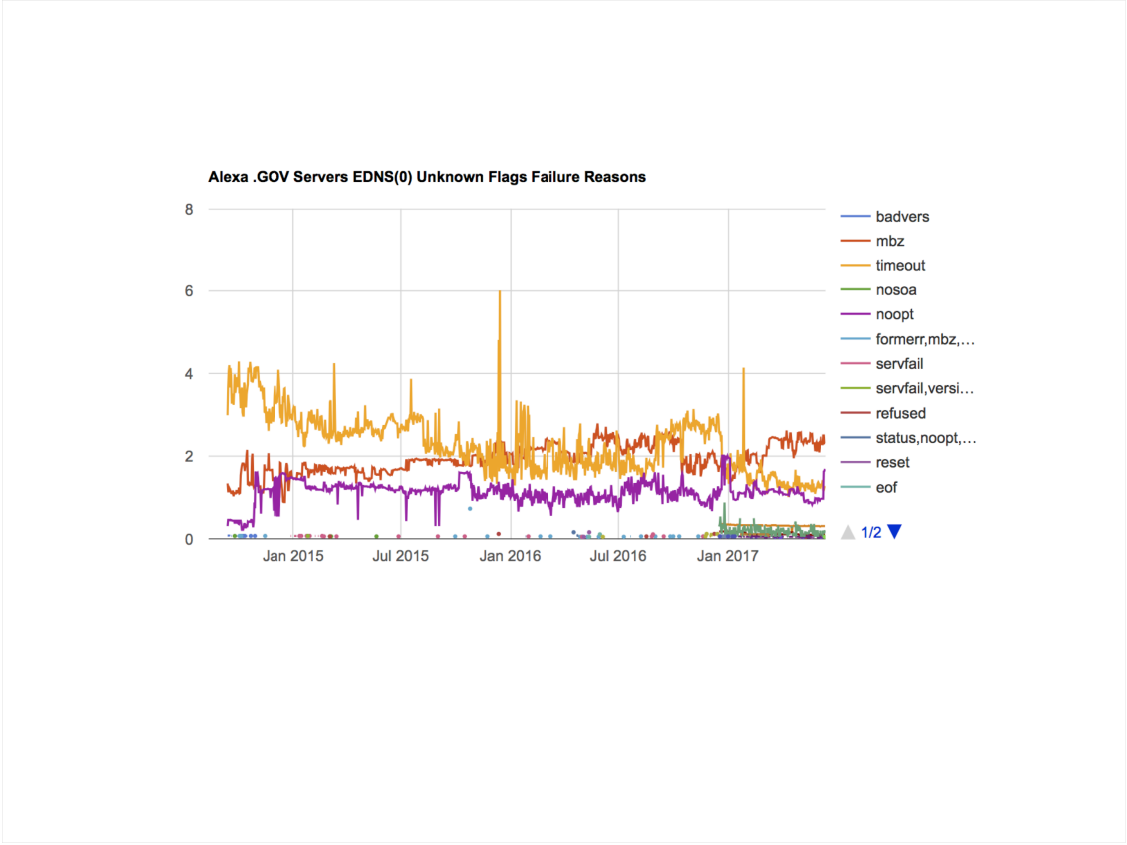
The green marked line indicates servers that incorrectly echo unknown EDNS option. Servers that do this are one of the reasons that EDNS Client Subnet is only supposed to be sent to white listed servers.

The presence of servers like these impact on how future EDNS options are designed.



The blue columns are correct responses to queries with unknown EDNS flags present. These are supposed to be ignored by EDNS servers.


Percentage of EDNS aware servers that handled unknown EDNS(0) flags correctly
100
95
90
85
80
Jan 2015
Jul 2015
Jan 2016
Jul 2016
Jan 2017
Root and TLD Servers
Alexa Top 1000 Servers
Alexa Bottom 1000 Servers
Alexa .GOV Servers
Alexa .AU Servers

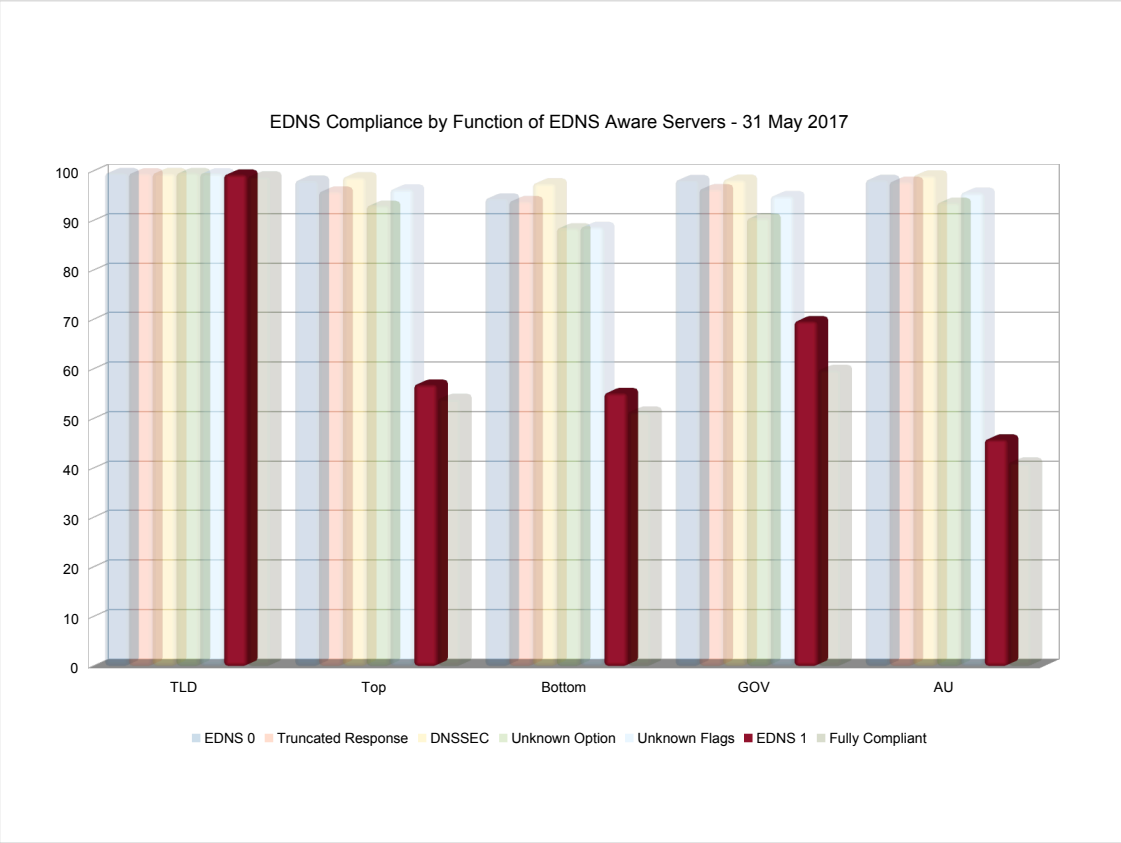


Unknown flags are mishandled in two ways.

The flag is echoed back rather than being ignored by the serve and firewalls block queries with unknown flags.

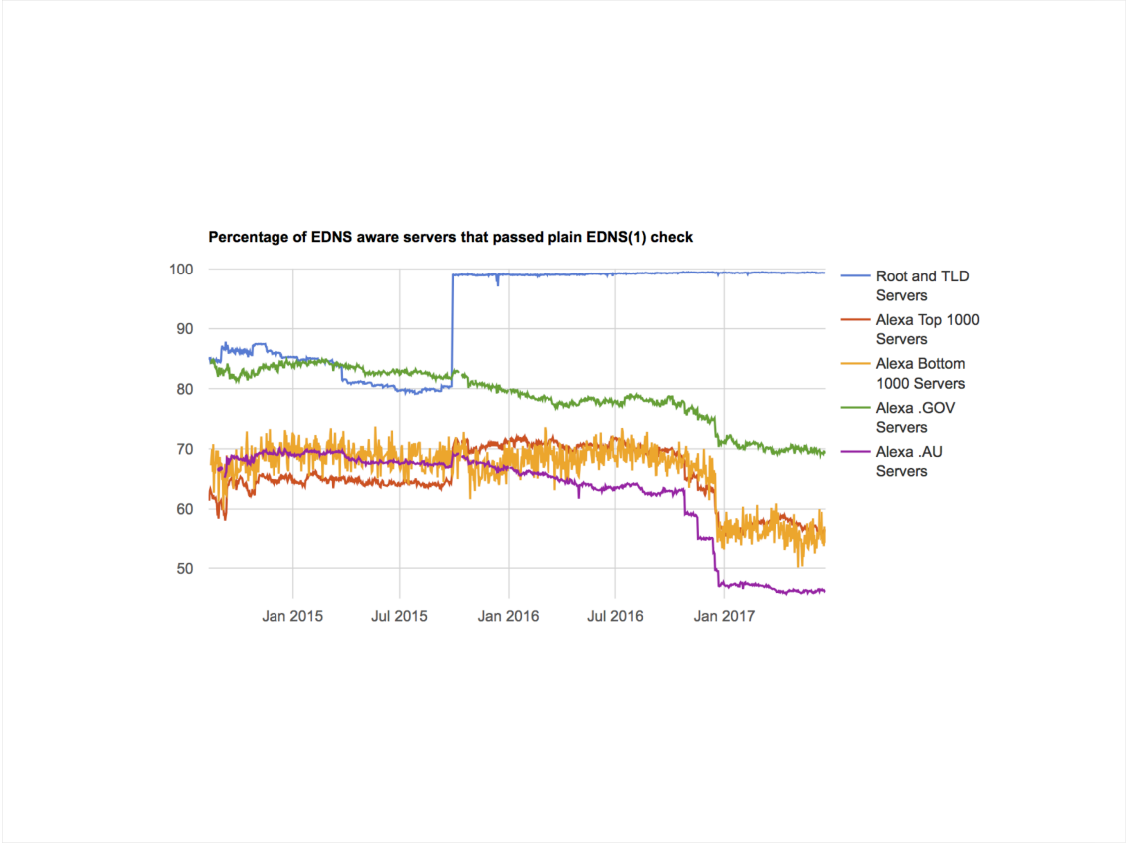
The echoing back of unknown flags means that you can't trust the presence of the flag in the response to mean anything. AD suffers from this at present.

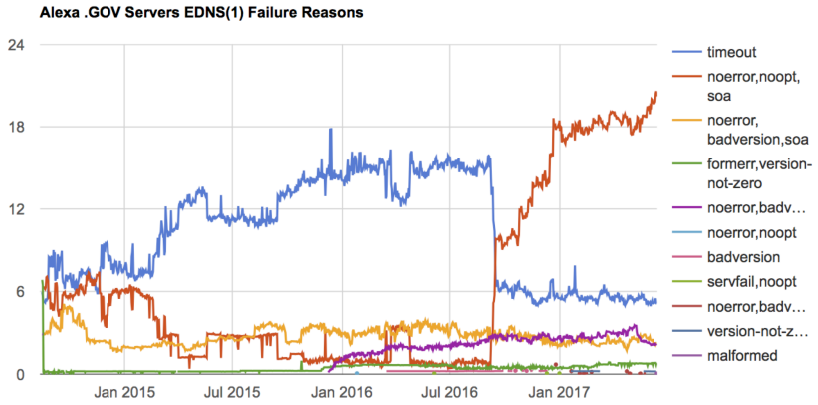
Blocking queries with a unknown flag will impact on DNSSEC validation as the resolver cannot determine if the query is being block because of the flag being present, if it is because the query is a EDNS query or because of packet loss.

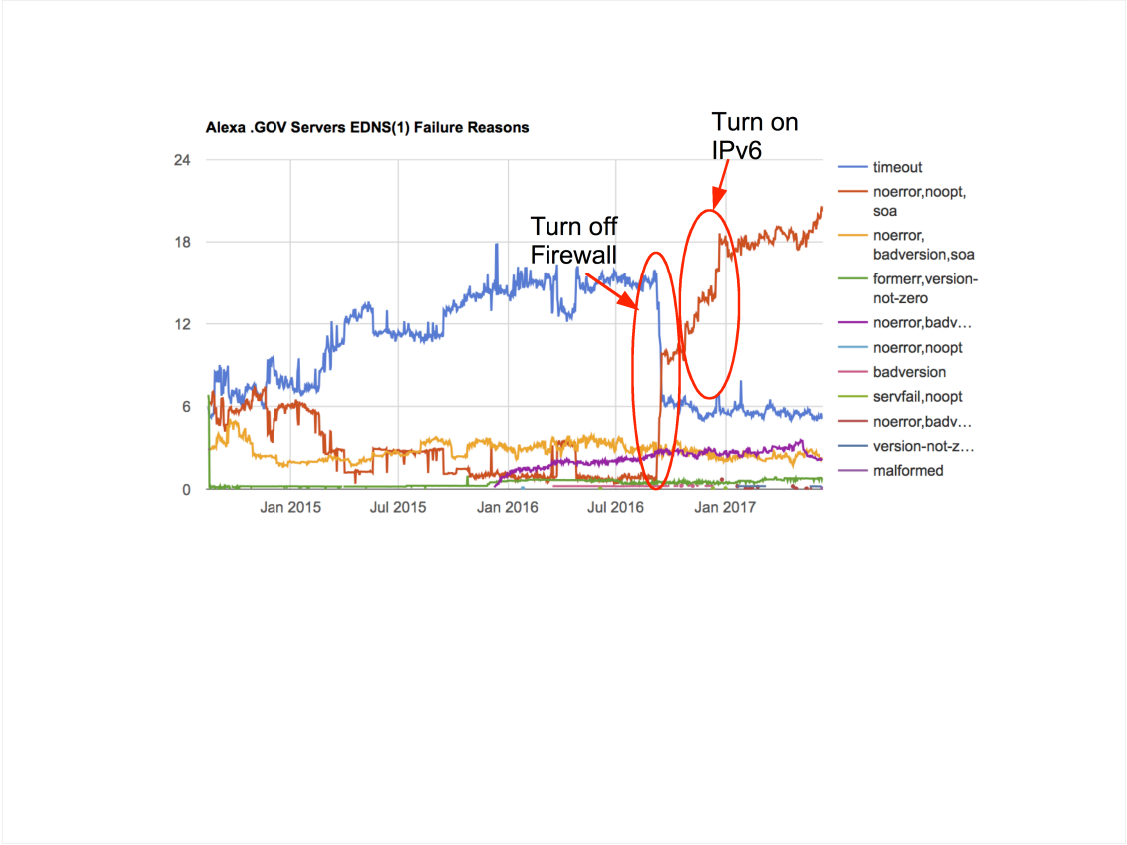


The red columns are servers that correctly answered EDNS(1) queries. Queries with unsupported EDNS versions are supposed to be responded to with rcode BADVERS and the version field set to the highest EDNS version supported by the server.

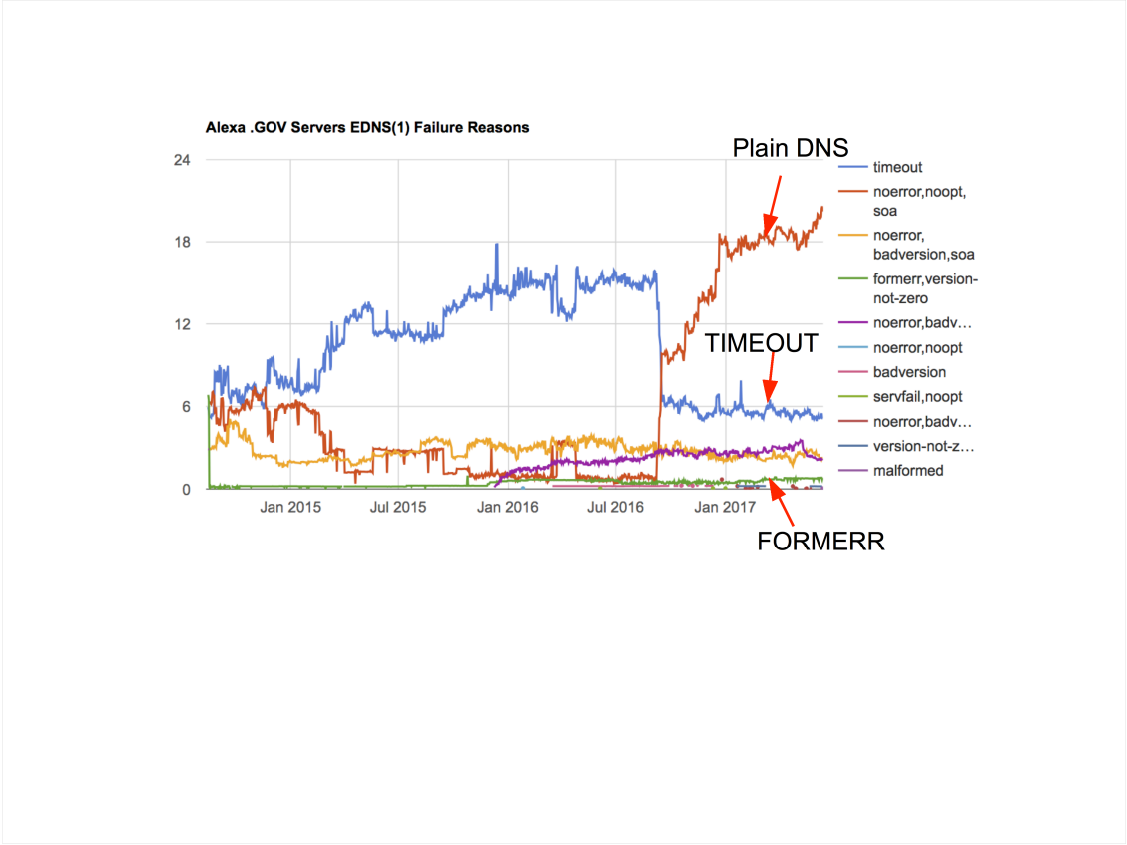
22/06/2017 17:37




Alexa .GOV Servers EDNS(1) Failure Reasons
24
18
12
6
0
Jan 2015
Jul 2015
Jan 2016
Jul 2016
Jan 2017
timeout
noerror,noopt, soa
noerror, badversion,soa
formerr,version-not-zero
noerror,badv…
noerror,noopt
badversion
servfail,noopt
noerror,badv…
version-not-z…
malformed



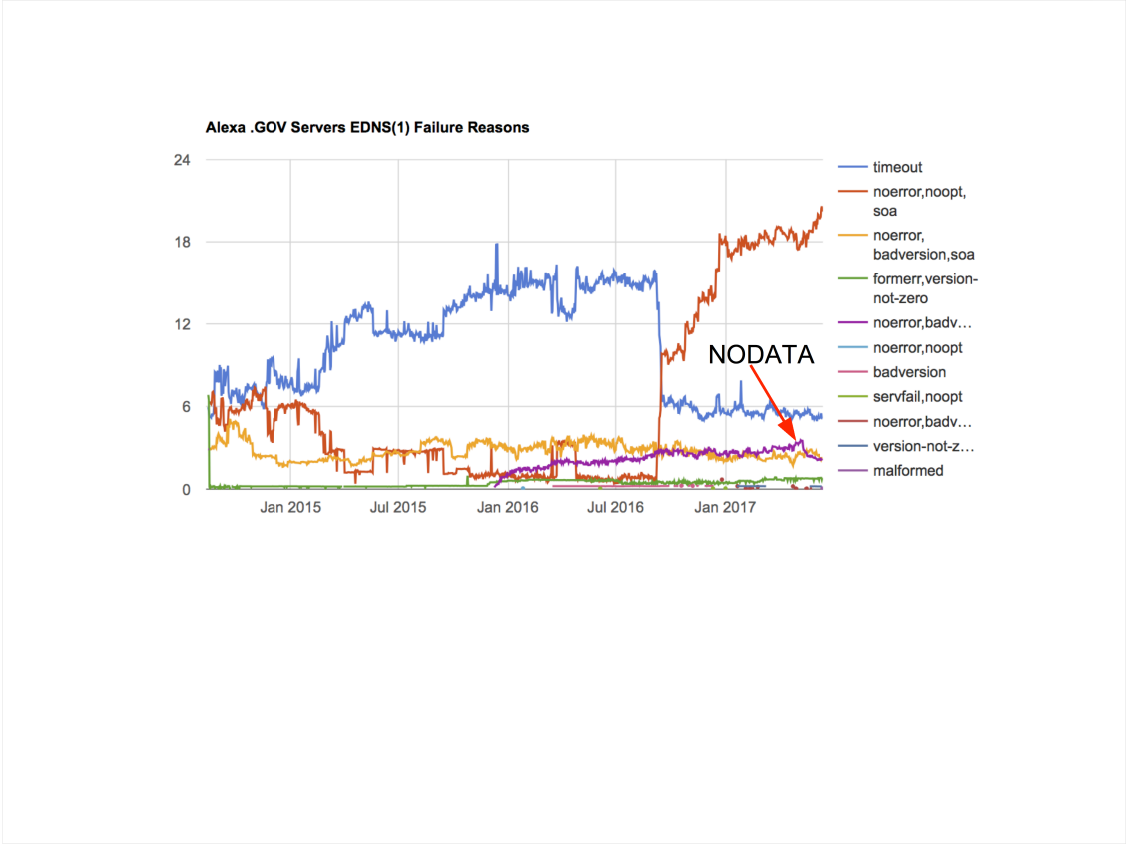
The step changes in the blue timeout line and the red no OPT record line show a large DNS hoster turning off the firewall in front of the DNS servers exposing the mishandling of EDNS queries by them. They then enabled IPv6 on those servers which is visible in the four step jumps as each region in turn was on.



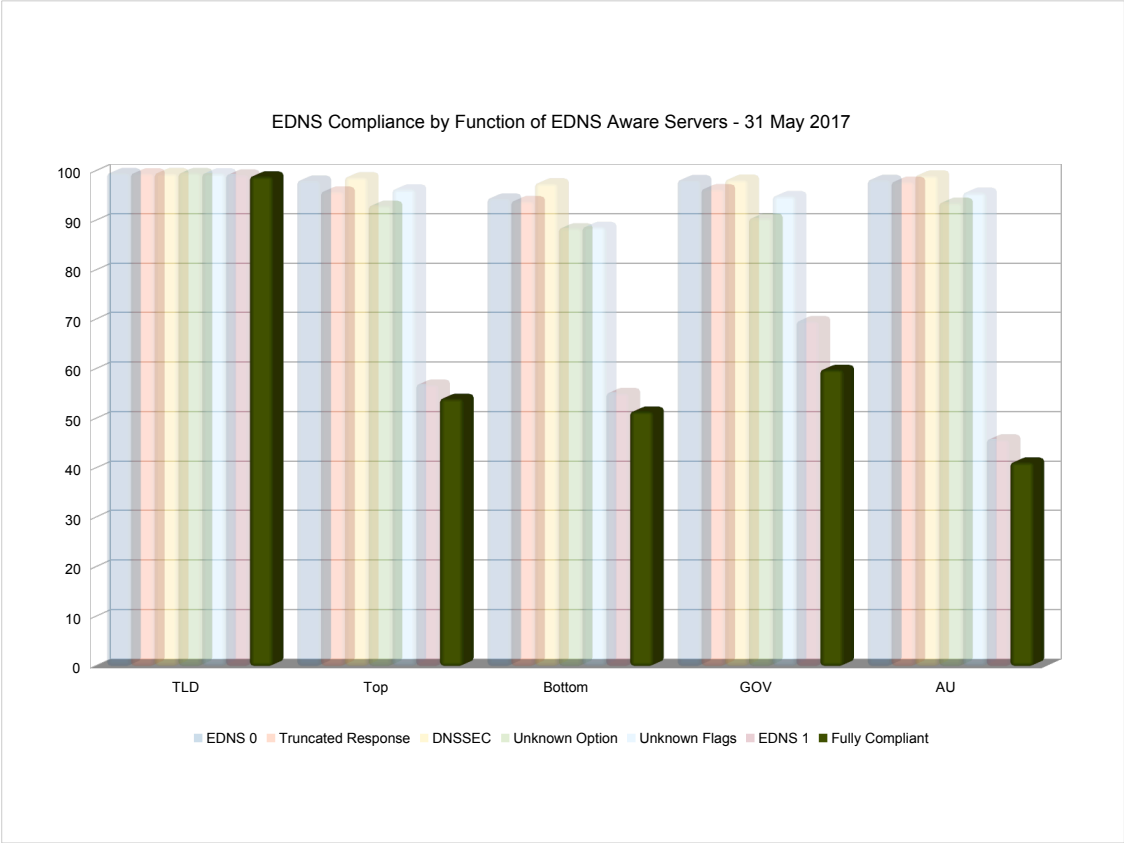
These three lines show responses that would result in named being unable to validate secure zones served by these servers successfully if there was ever a reason to send EDNS version 1 queries other than for testing purposes.

Plain DNS responses are incompatible with DNSSEC and are indistinguishable from those sent by servers that don't support EDNS at all.

Timeout and FORMERR will cause the server to re-try using plain DNS which is incompatible with DNSSEC.

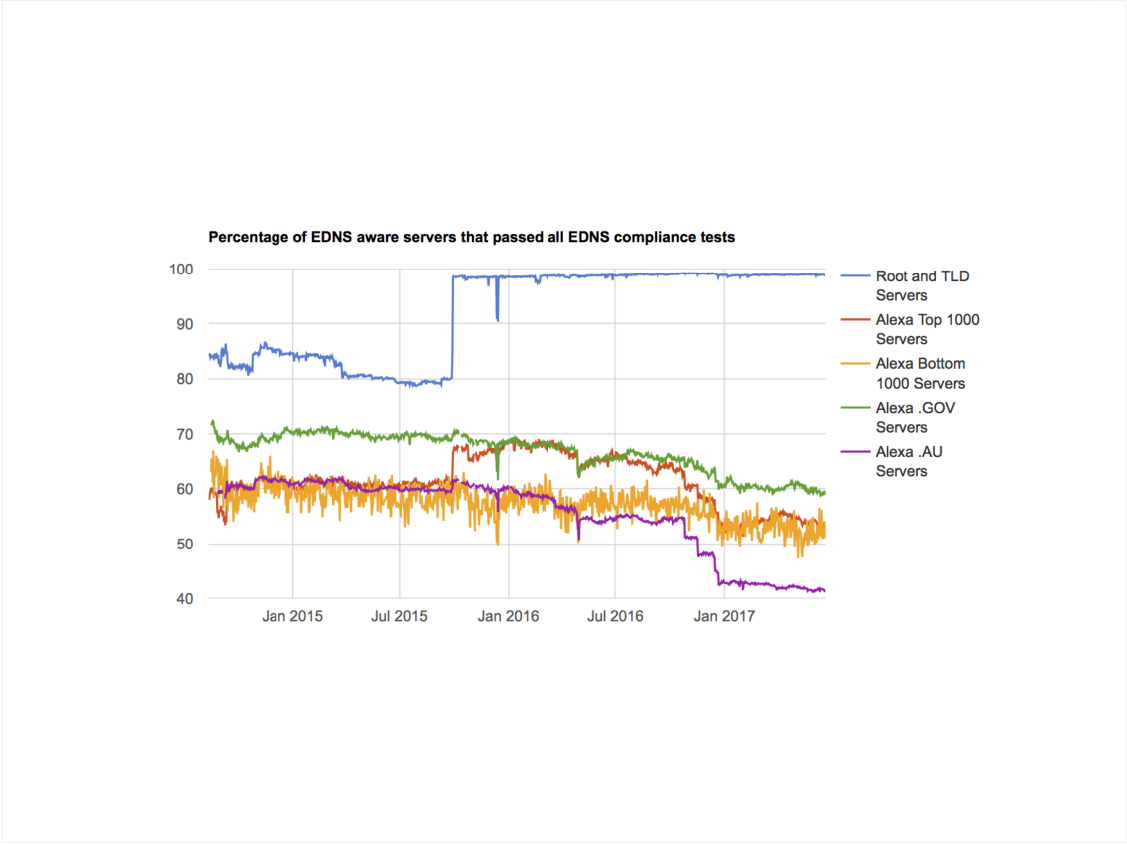


This purple line represents servers that incorrectly return responses that could be interpreted as NOERROR NODATA unless the EDNS version field and rcode field are sanity checked. The vendor of these servers has been informed of the issue.



The green columns show server which correctly responded to all EDNS extension mechanisms.

22/06/2017 17:37



Fixing Non-compliance

- Fix the DNS server implementations
- Fix firewall implementations
- Have agreed tests for non-compliance
- Introduce policy to say than non-compliant servers are not permitted.
- Introduce the new policy with grace period for existing servers
- Regularly test for compliance and remove delegations with non-complying servers

Fixing Non-compliance

- Fix the DNS server implementations
- Fix firewall implementations
- Have agreed tests for non-compliance
- Introduce policy to say than non-compliant servers are not permitted.
- Introduce the new policy with grace periods for existing servers and initially warnings for new servers
- Regularly test for compliance and remove delegations with non-complying servers

More Information

https://ednscomp.isc.org/

Test your own servers

https://ednscomp.isc.org/ednscomp

draft-ietf-dnsop-no-response-issue