



 lieter\_  
 PowerDNS

pieterlexis   
PowerDNS 

## IS YOUR DNS SERVER UP-TO-DATE?

---

Pieter Lexis – Senior PowerDNS Engineer  
April 22<sup>nd</sup> 2018

What's all this about?

A DNS recap

What is EDNS?

Issues with EDNS on the internet

Forcing EDNS Compliance

What can you do?

# INTRODUCTION

---

## Pieter Lexis

- Senior PowerDNS Engineer
- Trained as a SysAdmin
- Writing C++ and Python for PowerDNS
- Test, build and packaging automation <sup>1</sup>

---

<sup>1</sup>Shameless plug: <https://repo.powerdns.com/>

# POWERDNS

AN  COMPANY

- Founded in 1999
- Open Source since 2002
- Part of Open-Xchange since 2015
- Commercial support, deployment, and development of
  - PowerDNS Open-Source products
  - “PowerDNS Platform”

WHAT'S ALL THIS ABOUT?

---

# WHAT?

Coordinated effort by Open Source DNS vendors to

- Reduce code complexity
- Improve the health of the DNS on the Internet

by removing work-arounds in resolvers to deal with broken EDNS implementations in authoritative name servers.

# WHO?

- Internet Systems Consortium (BIND)
- CZ.NIC (Knot Resolver)
- NLNetLabs (Unbound)
- PowerDNS (PowerDNS Recursor)





# WHY?

- DNS resolvers have become unwieldy
- Many “incompatible” name servers on the Internet
- Prelude to future, other work-around removals (?)

## A DNS RECAP

---

# THE DNS – IN A FEW MINUTES

- Original idea from 1979 (IEN 116)
- Standardized in 1983 (RFC 882, 883)
- Current standard from 1987 (RFC 1034, 1035)
- Over 92 RFCs, 1677 pages<sup>2</sup>

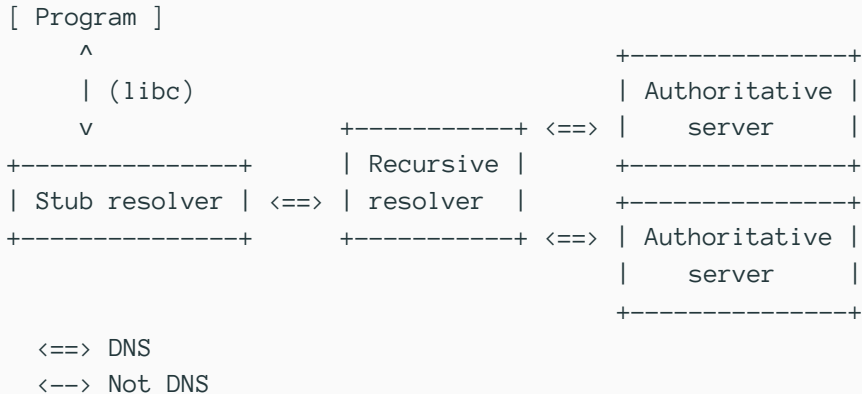
Want more? <https://powerdns.org/hello-dns/>

---

<sup>2</sup><https://powerdns.org/dns-camel/>

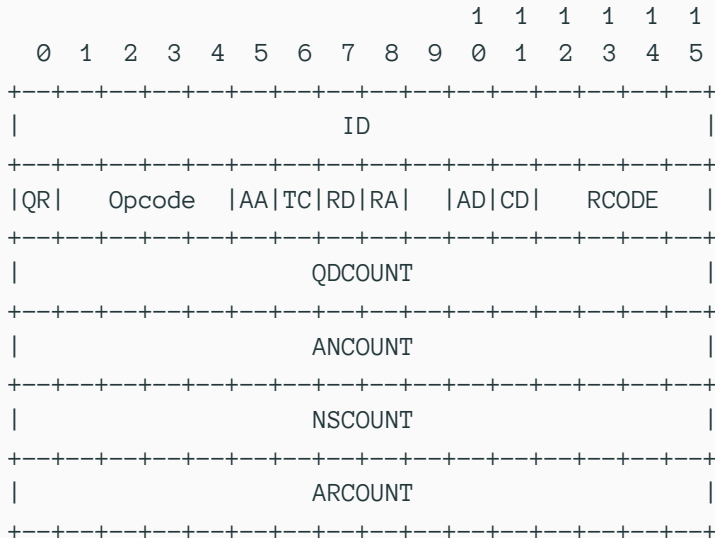
- Query-response protocol
- Maps hierarchical names to data
- Transport via UDP/53 and TCP/53 (other transports exist)
- Potentially the most fundamental Internet protocol

# THE DNS – MESSAGE FLOW



- Same packet structure for query and response
- A packet is (in order)
  - Header
  - Query section
  - Answer section
  - Authority section
  - Additional section

# THE DNS – PACKET HEADER



- NoError: “All is fine”
- FormErr: “You sent me garbage”
- ServFail: “I made a boo boo” or “I could not validate DNSSEC”
- NXDomain: “The queried name does not exist”
- NotImp: “I don’t know about that OPCODE”
- Refused: “I won’t do what you tell me”



- Only 16 possible RCODEs (11 assigned)
- One flag not defined
- No packets > 512 bytes on UDP
- No way to signal capabilities
- Packet format fixed (and rigid)

WHAT IS EDNS?

---

# Extension Mechanisms for DNS

From RFC 6891

“The Domain Name System’s wire protocol includes a number of fixed fields whose range has been or soon will be exhausted and does not allow requestors to advertise their capabilities to responders. This document describes backward-compatible mechanisms for allowing the protocol to grow.”

Network Working Group  
Request for Comments: 2671  
Category: Standards Track

P. Vixie  
ISC  
August 1999

Extension Mechanisms for DNS (EDNS0)

Internet Engineering Task Force (IETF)  
Request for Comments: 6891  
STD: 75  
Obsoletes: 2671, 2673  
Category: Standards Track  
ISSN: 2070-1721

J. Damas  
Bond Internet Systems  
M. Graff  
P. Vixie  
Internet Systems Consortium  
April 2013

Extension Mechanisms for DNS (EDNS(0))

- Requestor adds a record to additional section of query
- Responder sends FORMERR when not implemented
- Responder **MUST** send EDNS in response

- Name: Root domain (.)
- Type: OPT (41)
- Class: UDP buffer size
- TTL:
  - 8 bits: “Upper bits of the RCODE”
  - 8 bits: EDNS version (0)
  - 16 bits: Moar flags!
- RData: EDNS options

## Flag-based

- DNSSEC (RFC 4033, 4034, 4035 et al.)

## EDNS Options

Key-Value pairs with their own semantics.

Unknown options are ignored by responders.

- NSID (RFC 5001)
- Client Subnet (RFC 7871)
- DNS Cookies (RFC 7873)
- CHAIN Queries (RFC 7901)

## EDNS – WHAT DOES IT LOOK LIKE?

```
; <<>> DiG 9.12.1 <<>> @127.0.0.1 +dnssec www.powerdns.com
.....
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 61126
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags: do; udp: 4096
```



# EDNS – WHAT DOES AN OPTION LOOK LIKE?

```
; <<>> DiG 9.12.1 <<>> @pdns-public-ns2.powerdns.com +nsid +tries=1 +dnssec +
; (2 servers found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 56082
;; flags: qr aa; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags: do; udp: 1680
; NSID: 70 64 6e 73 2d 70 75 62 6c 69 63 2d 6e 73 32 2e 70
      6f 77 65 72 64 6e 73 2e 63 6f 6d ("pdns-public-ns2.powerdns.com")
```

# ISSUES WITH EDNS ON THE INTERNET

---

## Things seen

- Wrong RCODE (e.g. NOTIMP, SERVFAIL) on EDNS query
- No OPT in response (but no FORMERR)
- OPT record copied into response

“Responders that choose not to implement the protocol extensions defined in this document MUST respond with a return code (RCODE) of FORMERR to messages containing an OPT record in the additional section and MUST NOT include an OPT record in the response.”

## Things seen

- No response to EDNS query, response to non-EDNS query
- Packet truncated without TC flag
- EDNS in responses for non-EDNS requests

“Conformant middleboxes MUST NOT limit DNS messages over UDP to 512 bytes.”

## Things seen

- Unknown options are echoed back
- Version == 0 is ok, version != 0 does not send BADVERS
- Weird RCODEs or no response on unknown options

“If a responder does not implement the VERSION level of the request, then it MUST respond with RCODE=BADVERS.”

“Any OPTION-CODE values not understood by a responder or requestor MUST be ignored.”

- Workarounds are relatively easy
  - Retry other name server for zone
  - Retry without EDNS
- Which one to pick, when?
- How to detect what the issue was?

## FORCING EDNS COMPLIANCE

---

- DNS is complex
- Many interactions between features<sup>3</sup>
- Too much complexity in name servers already
- EDNS is 19 (or 6) years old

---

<sup>3</sup><https://blog.powerdns.com/2018/03/22/the-dns-camel-or-the-rise-in-dns-complexit/>



“All new releases of DNS software from CZ.NIC, ISC, NLnetlabs, and PowerDNS after February 1, 2019 will not contain code for the workaround of non-compliance problems with EDNS standard RFC 6891.”<sup>4</sup>

---

<sup>4</sup><https://en.blog.nic.cz/2018/03/14/together-for-better-stability-speed-and-further-extensibility-of-the-dns-ecosystem/>

WHAT CAN YOU DO?

---

# WHAT CAN YOU DO?

- Run up to date software!
  - Bind 9.12 and up
  - Knot (all versions)
  - NSD 1.0.0 and up
  - PowerDNS Authoritative Server 4.0 and up
- Check middleboxes and firewalls
- Test your domains

<https://ednscomp.isc.org/ednscomp/>

## EDNS Compliance Tester

Checking: 'loadays.org' as at 2018-04-17T21:06:10Z

loadays.org @149.202.166.43 (ns01.multihost.be.): edns=ok edns1=ok edns@512=ok ednsopt=ok edns1opt=ok do=ok ednsflags=ok docookie=ok edns@512tcp=ok optlist=ok

loadays.org @37.187.243.140 (ns03.multihost.be.): edns=ok edns1=ok edns@512=ok ednsopt=ok edns1opt=ok do=ok ednsflags=ok docookie=ok edns@512tcp=ok optlist=ok

**All Ok**

### Codes

- *ok* - test passed.

## EDNS Compliance Tester

Checking: 'google.com' as at 2018-04-17T21:23:04Z

google.com @216.239.32.10 (ns1.google.com.): [edns=noopt edns1=status,noopt,soa edns@512=noopt ednsopt=noopt edns1opt=status,noopt,soa do=noopt ednsflags=noopt docookie=noopt edns@512tcp=noopt](#) optlist=subnet

google.com @2001:4860:4802:32::a (ns1.google.com.): [edns=noopt,ipv6 edns1=status,noopt,soa edns@512=noopt ednsopt=noopt edns1opt=status,noopt,soa do=noopt ednsflags=noopt docookie=noopt edns@512tcp=noopt](#) optlist=subnet

google.com @216.239.34.10 (ns2.google.com.): [edns=noopt edns1=status,noopt,soa edns@512=noopt ednsopt=noopt edns1opt=status,noopt,soa do=noopt ednsflags=noopt docookie=noopt edns@512tcp=noopt](#) optlist=subnet

google.com @2001:4860:4802:34::a (ns2.google.com.): [edns=noopt,ipv6 edns1=status,noopt,soa edns@512=noopt ednsopt=noopt edns1opt=status,noopt,soa do=noopt ednsflags=noopt docookie=noopt edns@512tcp=noopt](#) optlist=subnet

google.com @216.239.36.10 (ns3.google.com.): [edns=noopt edns1=status,noopt,soa edns@512=noopt ednsopt=noopt edns1opt=status,noopt,soa do=noopt ednsflags=noopt docookie=noopt edns@512tcp=noopt](#) optlist=subnet

google.com @2001:4860:4802:36::a (ns3.google.com.): [edns=noopt,ipv6 edns1=status,noopt,soa edns@512=noopt ednsopt=noopt edns1opt=status,noopt,soa do=noopt ednsflags=noopt docookie=noopt edns@512tcp=noopt](#) optlist=subnet

google.com @216.239.38.10 (ns4.google.com.): [edns=noopt edns1=status,noopt,soa edns@512=noopt ednsopt=noopt edns1opt=status,noopt,soa do=noopt ednsflags=noopt docookie=noopt edns@512tcp=noopt](#) optlist=subnet

google.com @2001:4860:4802:38::a (ns4.google.com.): [edns=noopt,ipv6 edns1=status,noopt,soa edns@512=noopt ednsopt=noopt edns1opt=status,noopt,soa do=noopt ednsflags=noopt docookie=noopt edns@512tcp=noopt](#) optlist=subnet

## Checking: 'mateksys.com' as at 2018-04-09T19:47:11Z

mateksys.com @106.11.211.53 (dns7.hichina.com.): edns=ok edns1=timeout edns@512=ok ednsopt=timeout edns1opt=timeout do=ok ednsflags=ok doco  
edns@512tcp=timeout optlist=timeout  
mateksys.com @106.11.141.123 (dns7.hichina.com.): edns=ok edns1=timeout edns@512=ok ednsopt=timeout edns1opt=timeout do=ok ednsflags=ok doco  
edns@512tcp=timeout optlist=timeout  
mateksys.com @106.11.141.113 (dns7.hichina.com.): edns=ok edns1=timeout edns@512=ok ednsopt=timeout edns1opt=timeout do=ok ednsflags=ok doco  
edns@512tcp=timeout optlist=timeout  
mateksys.com @140.205.81.13 (dns7.hichina.com.): edns=ok edns1=timeout edns@512=ok ednsopt=timeout edns1opt=timeout do=ok ednsflags=ok doco  
edns@512tcp=timeout optlist=timeout

## CONCLUSION

---



# IN CONCLUSION

- EDNS is an enabler technology
- Some name servers do the wrong thing
- Open Source implementers got tired of workarounds
- These will be removed starting February 2019

# Test your domains

Keep your name servers up to date

Questions?

Thanks!