ECDSA P-256 support in DNSSEC-validating Resolvers

Geoff Huston, George Michaelson

APNIC Labs

October 2014

ECDSA

- Elliptic Curve Cryptography allows for the construction of “strong” public/private key pairs with key lengths that are far shorter than equivalent strength keys using RSA

  “256-bit ECC public key should provide comparable security to a 3072-bit RSA public key” *

- And the DNS protocol has some sensitivities over size
  - UDP fragmentation has it’s issues in V4 and V6

* http://en.wikipedia.org/wiki/Elliptic_curve_cryptography

So lets use ECDSA for DNSSEC

- Yes?
- Or maybe that's a Bad Idea!

- Is ECDSA a "well supported" crypto protocol?
- If you signed using ECDSA would they validate it?

The Test Environment

We used the Google Ad network to deliver a set of DNS tests to clients to determine whether (or not) they use DNSSEC validating resolvers

We used 4 tests:

1. no DNSSEC-signature at all
2. DNSSEC signature using RSA-based algorithm
3. DNSSEC signature using broken RSA-based algorithm
4. DNSSEC signature using ECDSA P-256 algorithm

The Test Environment

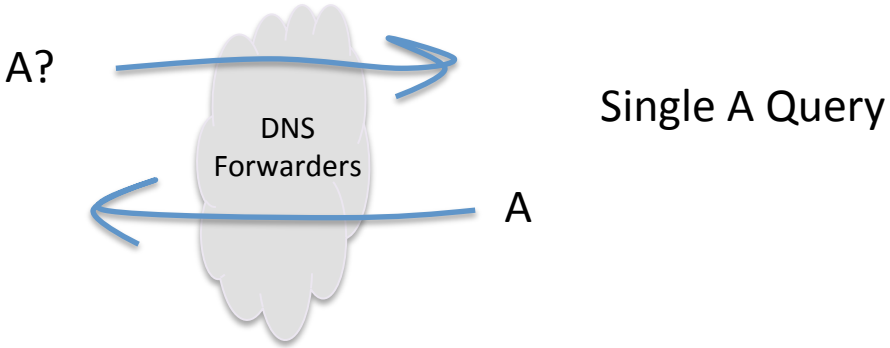
d.t10000.u2045476887.s1412035201.i5053.vne0001.4f167.z.dashnxdomain.net	Unsigned
e.t10000.u2045476887.s1412035201.i5053.vne0001.4f167.z.dotnxdomain.net	RSA Signed
f.t10000.u2045476887.s1412035201.i5053.vne0001.4f168.z.dotnxdomain.net	RSA signed (Badly)
g.t10000.u2045476887.s1412035201.i5053.vne0001.4f167.y.dotnxdomain.net	ECDSA-Signed

Mapped to a wildcard in the zone file

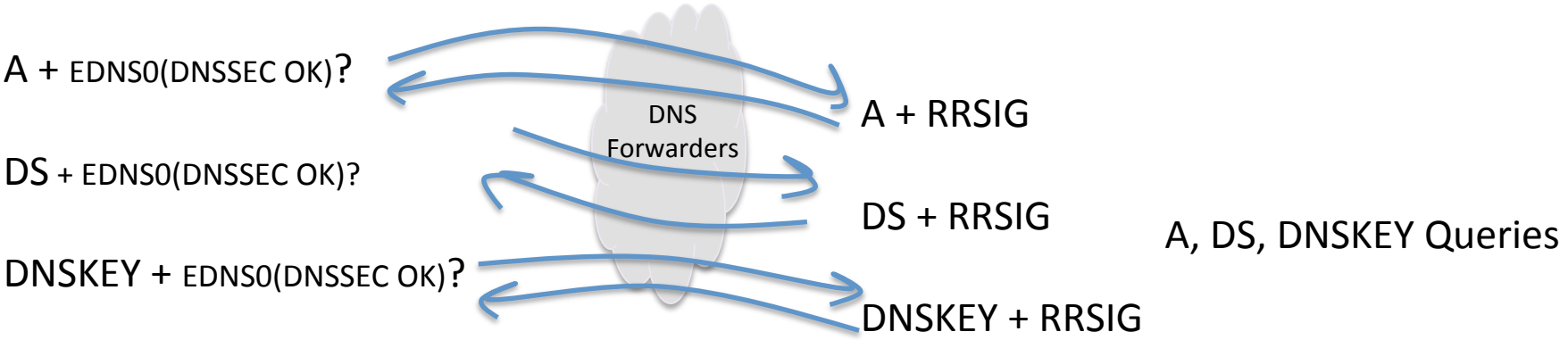
Unique Signed Zone

A Naïve View

A non-DNSSEC-validating resolver query:



A DNSSEC-Validating resolver query:



Theory: DNSSEC Validation Queries

`e.t10000.u2045476887.s1412035201.i5053.vne0001.4f167.z.dotnxdomain.net`

Query for the A resource record with EDNS0, DNSSEC-OK

query: e.t10000.u204546887.s1412035201.i5053.vne0001.4f167.z.dotnxdomain.net IN A +ED

Query the parent domain for the DS resource record

query: 2f7b3.z.dotnxdomain.net): query: 4f167.z.dotnxdomain.net IN DS +ED

Query for the DNSKEY resource record

query: 2f7b3.z.dotnxdomain.net): query: 4f167.z.dotnxdomain.net IN DNSKEY +ED

Practice: The DNS is “messy”

- Clients use multiple name servers, and use local timeouts to repeat the query
- Resolvers may use server farms, so that queries from a common logical resolution process may be presented to the authoritative name server from multiple resolvers, and each resolver may present only a partial set of validation queries
- Resolvers may use forwarding resolvers, and may explicitly request checking disabled to disable the forwarding resolver from performing validation itself
- Clients and resolvers have their own independent retry and abandon timers

First Approach to answering the ECDSA question – Statistical Inference

- A DNSSEC-aware resolver encountering a RR with an attached RRSIG that uses a known algorithm will query for DS and DNSKEY RRs

- A DNSSEC-aware resolver encountering a RR with an attached RRSIG that uses an unknown/unsupported crypto algorithm appears ***not*** to query for the DNSKEY RRs

Results

Over 22 days in September 2014 we saw:

3,773,420 experiments

937,166 experiments queried for the DNSKEY RR of a validly signed (RSA) domain (24.8%)

629,726 experiments queried for the DNSKEY RR of a validly signed (ECC) domain (16.6%)

Results

Over 22 days in September 2014 we saw:

3,773,420 experiments

937,166 experiments queried for the DNSKEY RR of a validly signed (RSA) domain (24.8%)

629,726 experiments queried for the DNSKEY RR of a validly signed (ECC) domain (16.6%)

If we assume that the DNSKEY query indicates that the resolver "recognises" the protocol, then it appears that there is a fall by 8.2% in validation when using the ECDSA protocol

1 in 3 experiments that fetched the DNSKEY in RSA did not fetch the ECDSA-signed DNSKEY

Hmmm

- How does this relate to affected users?
- How do validating resolvers manage an unrecognised algorithm failure?

- Lets try again and look at both DNS query and web log data

DNS resolver failure modes for an unknown signing algorithm

If a DNSSEC-Validating resolver receives a response RRSIG with an unknown crypto algorithm does it:

- ❑ Immediately stop resolution and return a status code of SERVFAIL?
- ❑ Fetch the DS RR and then return a status code of SERVFAIL?
- ❑ Fetch the DS and DNSKEY RRs and return a status code of SERVFAIL?
- ❑ Abandon validation and just return the unvalidated query result?

Second Approach to answering the ECC question – DNS + WEB

Data collection: 10/9/14 – 4/10/14

552,104 clients who appear to be exclusively using RSA DNSSEC-Validating resolvers

ECC Results:

Success:	76.45%	361,698	Saw fetch of the DNSSEC RRs and the URL

Fetched the URL but appeared not to validate

Failure (1)	19.64%	108,411	Did **not** see query of DNSKEY, but fetched the URL
Failure (2)	1.47%	8,121	Saw only A queries, but fetched the URL
Failure (3)	0.84%	4,615	Saw queries with DO set and not set, fetched the URL

Did **not** fetch the URL

Failure (4)	1.07%	5,927	Saw query of the DNSSEC RRs, NOT URL
Failure (5)	0.34%	1,875	Saw query of A, DS, not DNSKEY, NOT URL
Failure (6)	0.12%	655	Saw only A queries, NOT URL
Failure (7)	0.08%	436	Saw queries with DO set and not set, NOT URL

Apparent Fail:	**23.55%**	**130,040**

Results

- These results show that 76% of clients who appeared to exclusively use RSA DNSSEC-Validating resolvers were also seen to perform validation using ECDSA
- 22% of the the remaining clients fetched the object, even though the DNS queries showed that there was not a complete DNSSEC validation pass being performed
- Just 1.6% of clients did NOT fetch the URL

What? Really?

23.6% ECDSA validation failure is very surprising

- Don't forget that the subsection of users' resolvers being polled here already did RSA validation and appeared to correctly return SERVFAIL when the DNSSEC crypto was broken

The fact that most of the failures result in a fetch of the URL is even more surprising

- The expectation was that we would see far more SERVFAIL and far higher URL fail-to-fetch rates
- It seems that the resolvers involved in this behaviour appear to be tagging the domain as "not validatable" and passing back an "insecure" outcome

Where?

ECDSA failure rates – the % of users in each country who use RSA DNSSEC validating resolvers, but fail to validate when the DNSSEC crypto algorithm is ECC. Top 24 countries, ranked by Observed ECC Validation failure rates

1	MN	96.82	Mongolia
2	MT	96.68	Malta
3	FI	95.75	Finland
4	AD	93.41	Andorra
5	CY	92.61	Cyprus
6	BB	90.59	Barbados
7	FJ	89.93	Fiji
8	ZA	85.94	South Africa
9	AG	84.51	Antigua and Barbuda
10	LU	83.28	Luxembourg
11	AU	79.93	Australia
12	SI	79.51	Slovenia
13	NO	78.91	Norway
14	LY	77.13	Libya
15	YE	75.81	Yemen
16	GR	69.64	Greece
17	KW	68.69	Kuwait
18	RW	66.67	Rwanda
19	BY	63.38	Belarus
20	UA	62.15	Ukraine
21	KE	60.57	Kenya
22	BA	56.35	Bosnia and Herzegovina
23	JP	56.06	Japan
24	KZ	49.50	Kazakhstan

Who?

ECDSA failure rates – the % of users in each AS who use RSA DNSSEC validating resolvers, but fail to validate when the DNSSEC crypto algorithm is ECDSA – top 25 Ases ranked by ECC failure rate

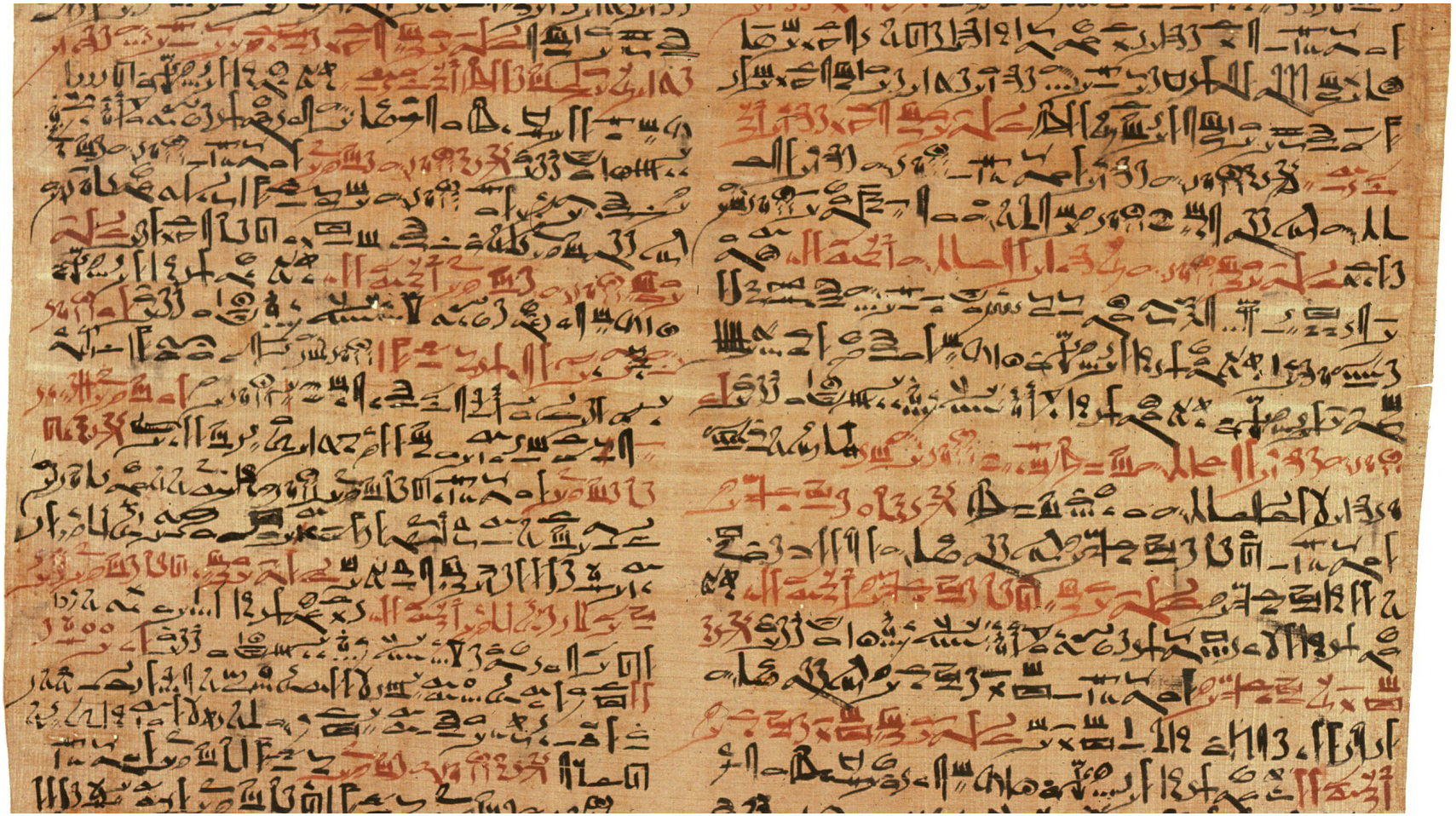
	AS	Fail Rate	Samples	AS Description
1	7155	100.00	202	WB-DEN2 - Viasat Communications Inc.,US
2	44143	100.00	662	VIPMOBILE-AS Vip mobile d.o.o.,RS
3	22363	100.00	157	PHMGMT-AS1 - Powerhouse Management, Inc.,US
4	12638	99.53	215	AS12638 E-Plus Mobilfunk GmbH & Co. KG,DE
5	33929	99.39	164	MASICOM-AS Telemach d.o.o.,SI
6	37457	99.36	933	Telkom-Internet,ZA
7	16014	99.25	398	EE-EMT AS EMT,EE
8	10219	99.17	362	SKYCC-AS-MAIN SKY C&C LLC,MN
9	7679	99.11	450	QTNET Kyushu Telecommunication Network Co.,Inc.,JP
10	1759	98.98	2,644	TSF-IP-CORE TeliaSonera Finland IP Network,FI
11	11815	98.97	291	Cooperativa Telefonica de V.G.G. Ltda.,AR
12	16232	98.79	1,238	ASN-TIM TIM (Telecom Italia Mobile) Autonomous System,IT
13	5603	98.77	5,039	SIOL-NET Telekom Slovenije d.d.,SI
14	17711	98.71	155	NDHU-TW National Dong Hwa University,TW
15	4804	98.70	1,456	MPX-AS Microplex PTY LTD,AU
16	12644	98.60	930	TELEMACH Telemach Autonomous System,SI
17	15735	98.58	1,059	DATASTREAM-NET GO p.l.c.,MT
18	53142	98.57	210	Friburgo Online LTDA ME,BR
19	41164	98.13	267	GET-NO GET Norway,NO
20	7992	97.94	679	COGECOWAVE - Cogeco Cable,CA
21	44489	97.31	335	STARNET Starnet s.r.o.,CZ
22	39651	96.82	943	COMHEM-SWEDEN Com Hem Sweden,SE
23	27813	96.70	485	Teledifusora S.A.,AR
24	47956	96.50	371	XFONE XFONE COMMUNICATION LTD,IL
25	52263	96.14	233	Telecable Economico S.A.,CR

Why?

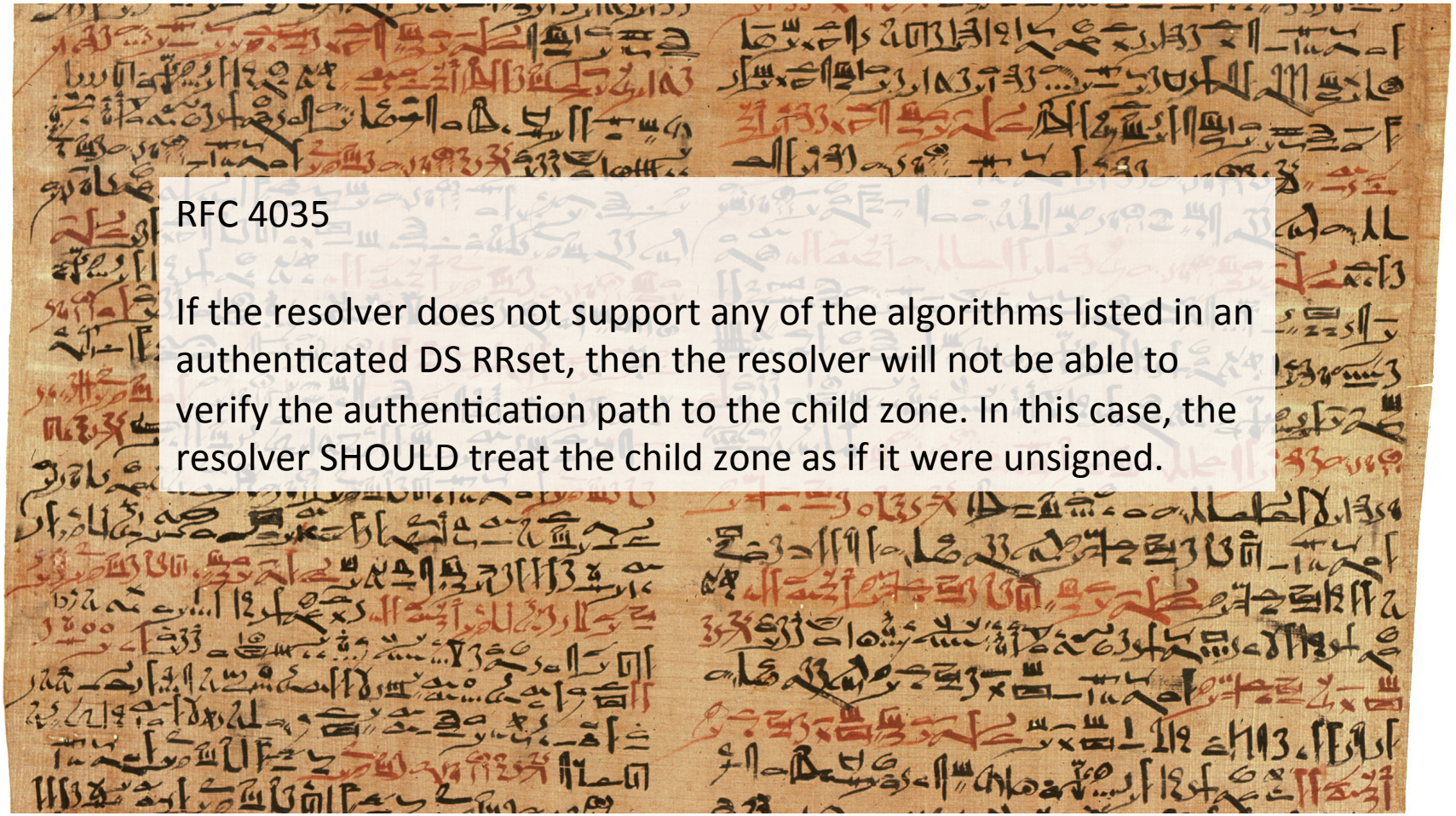
IPR issues:

- OpenSSL only added ECDSA support as from 0.9.8 (2005)
- Other bundles and specific builds added ECC support later
- Others still do not include ECC today

The Words of the Ancients



The Words of the Ancients



RFC 4035

If the resolver does not support any of the algorithms listed in an authenticated DS RRset, then the resolver will not be able to verify the authentication path to the child zone. In this case, the resolver SHOULD treat the child zone as if it were unsigned.

What About Google's Public DNS?

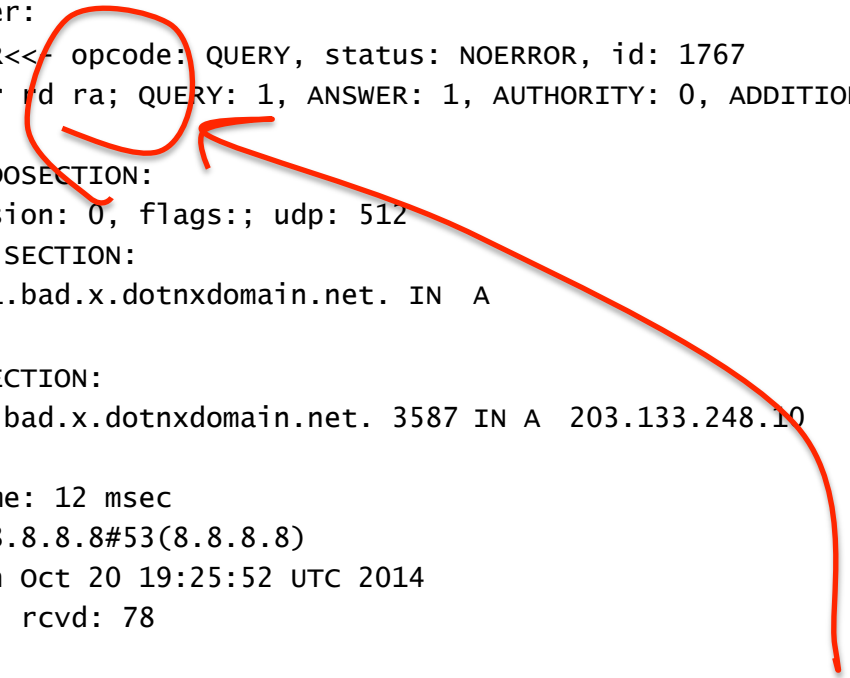
```
$ dig geoff.00001.bad.x.dotnxdomain.net @8.8.8.8

; <<>> DiG 9.9.5-P1 <<>> geoff.00001.bad.x.dotnxdomain.net @8.8.8.8
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 1767
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 512
;; QUESTION SECTION:
;geoff.00001.bad.x.dotnxdomain.net. IN  A

;; ANSWER SECTION:
geoff.00001.bad.x.dotnxdomain.net. 3587 IN A  203.133.248.10

;; Query time: 12 msec
;; SERVER: 8.8.8.8#53(8.8.8.8)
;; WHEN: Mon Oct 20 19:25:52 UTC 2014
;; MSG SIZE  rcvd: 78
```



The 'ad' flag is missing from the response!

If 8.8.8.8 does not validate ECDSA…

The level of support for the ECDSA algorithm in today's Internet is really very low indeed!

```
Data collection: 10/9/14 – 4/10/14

552,104 clients who appear to be exclusively using RSA DNSSEC-Validating resolvers

ECC Results:
  Success:       24.59%   130,220  Saw fetch of the DNSSEC RRs and the URL

Apparent Fail:   76.41%   421,884
```

Is ECDSA a viable crypto algorithm for DNSSEC?

If the aim is to detect efforts to compromise the DNS for the signed zone, then signing a zone with ECDSA limits the number of DNS resolvers who will validate the signature

Which is a shame, because the shorter key lengths could be attractive for DNS over UDP