A Survey of Authenticated Denial of Existence in DNSSEC

Shumon Huque

DNS-OARC 44 Workshop
February 7th 2025
Atlanta, GA, USA

Authenticated Denial of Existence

The mechanism by which DNSSEC proves that a domain name or record type doesn't exist.

Multiple scenarios:

- Proving that a domain name does not exist. (NXDOMAIN)
- Proving that a record type does not exist at a domain name. (NODATA)
  - A few special cases: proof of insecure delegation in a referral response
- Proving that there is a wildcard match for the queried name and type, and there is no closer match than the wildcard. (Wildcard)
- Proving that there is a wildcard match, but that the queried type does not explicitly exist. (Wildcard NODATA)

Authenticated Denial of Existence

DNSSEC is **data origin authentication**.

It is also **not a point to point protocol** – there may be many entities between a validating client/resolver and the publisher of the data, so transport (channel) protection is not sufficient to correctly authenticate responses.

Object security model required definition of a **new "data" record, NSEC** ("Next Secure") that chained together successive names in the zone, and asserted which record types were present at the first (owner) name.

Which implied a **Canonical Ordering of DNS names** in zone.

NSEC

Defined in the base DNSSEC specifications: RFC 4033, 4034, 4035

```
alfa.example.com.  86400 IN A 10.1.3.4
                   86400 IN MX 10 mail.example.com.
host.example.com.  86400 IN A 10.7.7.7
(RRSIG records omitted for brevity)

alfa.example.com.  86400 IN NSEC host.example.com. A MX RRSIG NSEC
```

NSEC NODATA response

```
$ dig +dnssec  jj.example.test. AAAA

;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 22185
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 4, ADDITIONAL: 1


;; AUTHORITY SECTION:
jj.example.test.   3600 IN      NSEC    nn.example.test. A RRSIG NSEC
jj.example.test.   3600 IN      RRSIG   NSEC <rdata omitted>
```

NSEC matching query name

Type Bitmaps show that data for the query type (AAAA) doesn't exist

NSEC NXDOMAIN response

NSEC covering wildcard
**.example.com*

```
$ dig +dnssec  kk.example.test. A

;; ->>HEADER<<- opcode: QUERY, status: NXDOMAIN, id: 49728
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 6, ADDITIONAL: 1

;; AUTHORITY SECTION:
example.test.      3600 IN  NSEC    bb.example.test. A NS SOA RRSIG NSEC DNSKEY
example.test.      3600 IN  RRSIG    NSEC <rdata omitted>

jj.example.test.   3600 IN  NSEC    nn.example.test. A RRSIG NSEC
jj.example.test.                3600 IN      RRSIG    NSEC <rdata omitted>
```

NSEC covering query name

NSEC3

RFC 5155: DNSSEC Hashed Authenticated Denial of Existence

Addressed 2 limitations in NSEC:

- Zone Enumeration vulnerability.
- Lack of ability to incrementally deploy DNSSEC in large delegation centric zones with sparsely signed child zones.

Political marriage of proposals to address both of these (NSEC2 and [*] Opt-In), neither of which by themselves had enough support in the IETF, but together, they were refined and produced NSEC3.

[*] Opt-In was actually preceded by earlier work: DNSNR, which may have been the original basis.

NSEC3: Zone Enumeration Defense

NSEC violates an important security tenet - the **principle of minimum disclosure**.

NSEC is vulnerable to trivial zone enumeration, by successive queries to walk NSEC chain.

A complete zone can provide adversaries a lot of info, e.g., for reconnaissance, harvesting email addresses for spam, disclosure of information identifying customers, clients, assets etc

Many zone owners do not want bulk disclosure of zone data, & in some cases (e.g., some ccTLD registries) may be under legal obligation to protect such disclosure.

NSEC3 attempts to address this problem, by **chaining together cryptographic (SHA-1) hashes of the names in the zone**, rather than the actual names.

Raises the bar for zone enumeration; does not comprehensively prevent it.

Offline dictionary attacks are still possible (and often quite viable)

NSEC5

NSEC5 was **provably secure against zone enumeration**, by **replacing the NSEC3 hash with a verifiable random function** based on asymmetric key cryptography.

But it did not have enough traction in the IETF, and was abandoned (additional complexity, high adoption & migration cost, not on technical merits).

Some references for the curious:

- NSEC5: Provably preventing Zone Enumeration, Feb 2015
- Making NSEC5 Practical for DNSSEC, Feb 2017
- NSEC5 Presentation at DNS-OARC, May 2017

NSEC3: Opt-Out

“Opt-Out” flag in NSEC3 records that specifies that **insecure delegations to child zones within the NSEC3 span do not need to have signed NSEC records matching their names to cryptographically assert their security status**.

This allows addition & removal of such insecure delegations without creating or recalculating signed NSEC3 records & re-adjusting the NSEC3 chain, making it more efficient to maintain the zone computationally and in terms of storage. This was a real issue for very large delegation centric zones with a sparse number of signed children.

Has very specific applicability, and is not widely used or useful outside TLDs.

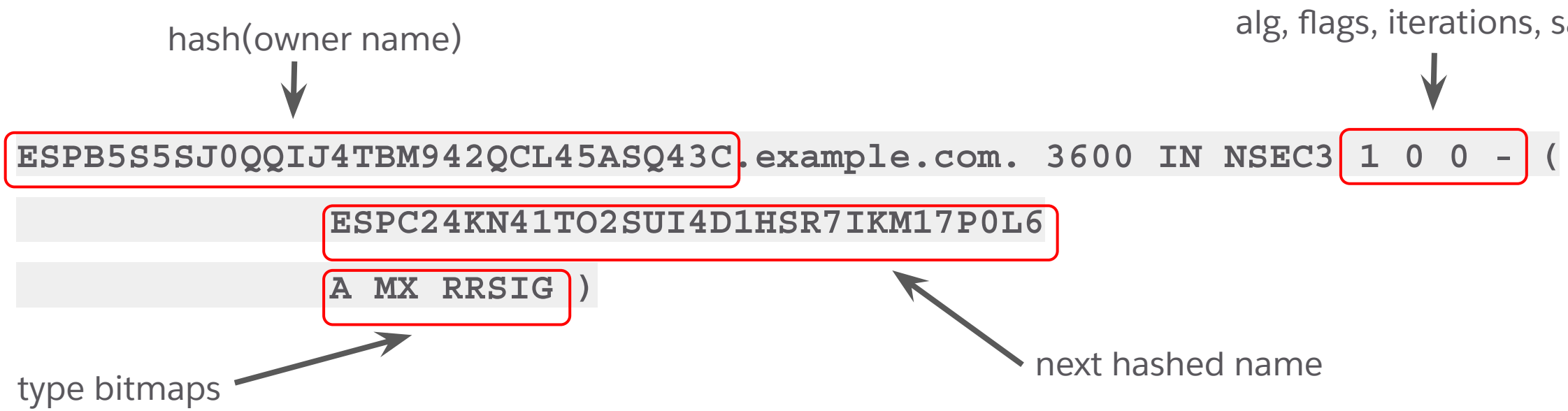
NSEC3

Compute NSEC3 hash of each name in the zone.

Sort the hashes into a chain.

Put consecutive hashes in NSEC3 records.

(Evolving DNSSEC guidance (RFC9276) now recommends not using a salt and setting the hash iteration count to 0.)



(Note: nsec3 hash as it appears in the owner name label and in the presentation format of the next hashed name is encoded in base32 with extended hex alphabet, which maintains the same sort order as the binary hash.)

NSEC3 NXDOMAIN response

- *bar.foo.jj.example3.test doesn't exist.*
- *The closest enclosing name that does exist is jj.example3.test*
- *Next closer name is foo.jj.example3.test*

```
$ dig +dnssec bar.foo.jj.example3.test. A
;; ->>HEADER<<- opcode: QUERY, status: NXDOMAIN, id: 41788

;; AUTHORITY SECTION:
NBAL9OU3DS4KFUR1HFJU58D6TBDOIV15.example3.test. 3600 IN NSEC3 1 0 0 - NN50FTK0UM
7O9TE2U3D2T4SVPKF2K5GH A RRSIG

JMPRR5B5RA7OKARLV5M3VMICCTRHDSMT.example3.test. 3600 IN NSEC3 1 0 0 - KPF2SQURFV
B81CVI4348L03BAQ23BET5 CNAME RRSIG

9TSHJ0PQ7RM48MHOHTMBBQCCHN8E3MKC.example3.test. 3600 IN NSEC3 1 0 0 - A7LLO9NV3L
U36V3JNECKCL52SSMHPDUV CNAME RRSIG
```

1. *Matches closest encloser*
2. *Covers next closer name*
3. *Covers wildcard at closest encloser*

```
$ nsec3hash -r 1 0 0 - jj.example3.test.
jj.example3.test. NSEC3 1 0 0 - NBAL9OU3DS4KFUR1HFJU58D6TBDOIV15

$ nsec3hash -r 1 0 0 - foo.jj.example3.test.
foo.jj.example3.test. NSEC3 1 0 0 - KIQLG6DNKL16F670VD0NK421UDU9FHP4

$ nsec3hash -r 1 0 0 - *.jj.example3.test.
*.jj.example3.test. NSEC3 1 0 0 - 9VKDUTEO5J6ADVTB3DP38RO0807A7O81
```

1. nsec3 hash of closest encloser
2. nsec3 hash of "next closer" name
3. nsec3 hash of wildcard at closest encloser

Pre-computed Signatures vs Online Signing

Classic NSEC/NSEC3 can support either:

Pre-computed signatures (where the signatures of all authoritative data in the zone are pre-computed, usually on a backend signing server, that is possibly offline or not queryable)

or

Online signing, where the signatures in the DNS response data are computed on the fly.

Pre-computed vs Online Signing tradeoffs

Pre-computed Signatures

More secure: Signing keys typically offline or on servers that do not accept network connections.

More performant: all the signatures are pre-calculated.

Zone databases are larger: they contain all the pre-computed signatures.

More zone maintenance and churn needed to refresh signatures regardless of whether zone data is queried. Plus zone transfer load.

Harder to support dynamically generated responses (e.g., traffic management etc)

Online Signing

Less secure: Signing keys need to be kept online at all the authoritative servers answering queries.

Less performant: signatures are computed in real time when answering queries.

Zone databases are smaller: they don't include any signatures.

No additional zone maintenance needed, though online signature caches (if implemented) may need some.

Can easily support dynamically generated responses.

“Minimally Covering” NSEC or NSEC3

A mode of NSEC/NSEC3 enabled by online signing, which uses epsilon functions to calculate minimal NSEC/3 ranges rather then using the actual before and after names in the zone.

Effectively prevents zone enumeration.

Easier for some implementations which lack zone data structures that permit them to efficiently compute predecessor and successor names.

Utilized by (and deployed in the field) by:

- NSEC and NSEC3 “White Lies”
- Compact Denial of Existence (formerly “Black Lies”)

NSEC White Lies

[RFC 4470: Minimally Covering NSEC Records and DNSSEC On-line Signing]

Dynamically constructing a minimally covering NSEC for the non-existent name "**foo.example.com**" - compute predecessor and successor names:

```
fon\255\255\255\255\255\255\255\255\255\255\255\255\255\255\255\255\255\255\255\255\2
55\255\255\255\255\255\255\255\255\255\255\255\255\255\255\255\255\255\255\255\255\25
5\255\255\255\255\255\255\255\255\255\255\255\255\255\255\255\255\255\255\255\255\255
.example.com. 3600 IN NSEC \000.foo.example.com. ( NSEC RRSIG )
```

```
fon\255.example.com. 3600 IN NSEC \000.foo.example.com. ( NSEC RRSIG )
```

```
fon~.example.com. 3600 IN NSEC foo!.example.com. ( NSEC RRSIG )
```

NSEC3 White Lies

Introduced in RFC 7129 (Feb 2014) - Authenticated Denial of Existence, although implementations like Phreebird (Dan Kaminsky) predated it.

Perfect epsilon function is easy to compute. The predecessor and successor names are just the NSEC3 hash of the name minus one and plus one.

```
$ nsec3hash -r 1 0 0 - bar.example.com
bar.example.com NSEC3 1 0 0 - OS1AK2KBC42BPD098IG1QJA8JMT8095E

    Predecessor: OS1AK2KBC42BPD098IG1QJA8JMT8095D (hash minus one)
    Successor:   OS1AK2KBC42BPD098IG1QJA8JMT8095F (hash plus one)

Resulting NSEC3 record that covers bar.example.com:

OS1AK2KBC42BPD098IG1QJA8JMT8095D.example.com. 3600 IN NSEC3 1 0 0 - (
                    OS1AK2KBC42BPD098IG1QJA8JMT8095F )
```

Compact Denial of Existence

Answers queries for a non-existent name by claiming that the name exists, but doesn't have any data associated with the queried record type!!

Such answers require just 1 NSEC record (compared to up to 2 NSEC or 3 NSEC3 records)

Minimize both message size and computational cost

Side Effect: Loss of visibility of the NXDOMAIN signal! (with many impacts)

Originally developed and deployed by Cloudflare in ~ 2016. Subsequent wider adoption in industry by other providers (NS1, Route53, Oracle)

Now being standardized & enhanced in the IETF. RFC to be published soon.

- NXNAME pseudo type to restore NXD visibility; signaled RCODE restoration, & extension to NSEC3

Compact Denial of Existence (cont.)

A response to a non-existent name will return a **NOERROR response code** with a dynamically constructed signed NSEC record matching the non-existent name, whose Type Bitmaps Field contains only “NSEC RRSIG NXNAME”.

NXNAME pseudo RR type is needed to precisely identify non-existent names and distinguish them from empty non-terminals. (Also new, and not yet universally adopted; CF and NS1 have so far).

There are no “covering” NSEC records, since everything is claimed to exist. Only the minimal next domain name has to be computed (prepend a zero octet label).

```
foo.example.com. 3600 IN NSEC \000.foo.example.com. RRSIG NSEC NXNAME
```

non-existent name

minimal next name

pseudo RR type to indicate non-existence

Hybrid Modes

In the same implementation/provider:

- In theory parts of the zone could use pre-computed NSEC/3, other parts could use online, with or without minimal spans, etc., but this is not commonly seen.

Across different providers:

Multi-Signer DNSSEC case - each provider may be using a different mode of authenticated denial of existence (pre-computed, online signing, online with minimally covering nsec).

Negative responses are self contained, so hybrid modes should work fine, though **combining different modes may negate the advantage bestowed by any given one** (e.g. zone enumeration protection, efficacy of aggressive negative caching, etc.)

Authoritative Server Side Implementations

Open Source Software

Method/Implementation	**BIND**	**NSD**	**PowerDNS**	**Knot**
NSEC Classic	✅	✅	✅	✅
NSEC3 Classic	✅	✅	✅	✅
NSEC White Lies				
NSEC3 White Lies			✅	
NSEC Compact Denial				✅
NSEC3 Compact Denial				

Authoritative Server Side Implementations

Commercial DNS Providers

Method/Implementation	**NS1**	**Cloudflare**	**UltraDNS**	**Google**	**Route53**	**Azure [2]**	**OCI**
NSEC Classic							
NSEC3 Classic				✅[1]			
NSEC White Lies			✅				
NSEC3 White Lies							
NSEC Compact Denial	✅	✅			✅	?	
NSEC3 Compact Denial							✅

[1] but apparently on-the-fly signing to accommodate dynamically computed answers
[2] Azure not tested yet, were on the verge of introducing their DNSSEC implementation in late 2024 as I was putting together this presentation.

Negative Response Synthesis (by Resolvers)

Negative responses:

- NXDOMAIN
- NODATA
- Wildcard matches

When resolvers are able to synthesize subsequent negative responses from a prior answer from an authority server, this has a beneficial effect of reducing overall load in the DNS - unnecessary outbound queries to authority servers are suppressed; and the resolver does not need to fetch these answers from those servers.

Can help mitigate a number of classes of attacks too, such as random subdomain attacks.

RFC 8020 NXDOMAIN Synthesis

[RFC 8020: NXDOMAIN: There Really Is Nothing Underneath]

Permits NXDOMAIN synthesis for both signed and unsigned responses.

But advises caution for the unsigned case: Spoofed responses can be employed by an adversary to prune out entire subtrees from a resolver's cache in one go, rather than RRset by RRset.

Note: Compact Denial of Existence precludes NXDOMAIN synthesis since no traditional NXDOMAIN responses are returned.

NXDOMAIN synthesis (8020)

Implementation	Unsigned	Signed
ISC BIND		✅
Unbound		✅
PowerDNS		✅
Knot DNS		✅
Google		✅
Cloudflare		
OpenDNS		
Route53		

RFC 8198 Aggressive Negative Caching

RFC 8198: Aggressive Use of a DNSSEC-validated Cache

Negative responses from an entire NSEC or NSEC3 span.

NXDOMAIN, NODATA, & Wildcard synthesis.

For NXDOMAIN, a superset of RFC8020 for signed answers.

Effectiveness:

- NSEC - clearly observed benefits seen in the field
- NSEC3 case more complex: Aggressive Negative Caching Effectiveness w/ NSEC3 - O. Moerbeek, Feb 2023, OARC40

Note: Zones that employ minimally covering NSEC/NSEC3 records negate effectiveness of ANC, because there is practically no usable NSEC/NSEC3 span.

ANC - Negative response synthesis

	NSEC			NSEC3		
Implementation	**NXDOMAIN**	**NODATA**	**Wildcard**	**NXDOMAIN**	**NODATA**	**Wildcard**
ISC BIND	✅	✅	✅			
Unbound	✅	✅	✅	✅		
PowerDNS	✅	✅	✅	✅	✅	✅
Knot DNS	✅	✅	✅	✅	✅	✅
Google	✅	✅	✅			
Cloudflare						
OpenDNS						
Route53						

Caveats: For some, configurable options; limitation to specific zones

Perils of Negative Response Synthesis

Broken authoritative servers

Example: <name> A gives answer, but <name> AAAA gives NXDOMAIN. What happens if you query the AAAA first, then the A?

Broken CDNs that answer NXDOMAIN for intermediate empty non-terminal names whose descendants clearly exist.

    Exposed by the deployment of Query-Name Minimization

Order of operations effects, complicating troubleshooting

Effects of Imprecise NSEC/NSEC3 bitmaps in responses:

- Operational Experiences with DNSSEC Signed Zones - S. Huque, Feb 2023, OARC40

A cursory point-in-time survey of Top Sites

Tranco[1] list of top 1 million sites on date 2024-12-20

DNSSEC itself is quite sparsely deployed below the Top Level Domains

We'll examine the NSEC/NSEC3 characteristics of the ~87K signed domains in the top million.

	#Signed	%Signed
Top 100	3	3.00%
Top 1,000	95	9.50%
Top 10,000	1091	10.91%
Top 100,000	10032	10.03%
Top 1,000,000	87733	8.77%

[1] Tranco: A Research-Oriented Top Sites Ranking Hardened Against Manipulation.

NSEC Feature Adoption in Top 1M

Examining 72,656 zones (after excluding ~ 15k zones that had broken DNSSEC) on 2024-12-20

NSEC	**43,339**		**59.50%**
Classic	10,462	24.20%	
White Lies	471	1.09%	
Compact Denial	32,296	74.71%	
NSEC3	**29,427**		**40.50%**
Classic	29,421	99.98%	
White Lies	0	0.00%	
Compact Denial	6	0.02%	

[There are also 2,289 NSEC3 zones using Opt-Out. For what benefit?]

Some Observations

- Some estimation involved in differentiating nsec types due to theoretical range of possible epsilon functions.
- Eliciting NXDOMAIN may require trial & error or multi-label fuzzing if there are wildcards directly under zone apex.
- Majority of zones employ some form of zone enumeration protection (NSEC3, NSEC White Lies, or Compact Denial of Existence).
- NSEC Feature Distribution is skewed by a few large providers, e.g. Cloudflare with Compact Denial - 3/4th of NSEC is CDOE.
- NSEC3 variant of Compact Denial, new & hitherto unseen, was discovered through this measurement.
- NSEC3 White Lies not observed

Top Level Domains (TLD) Survey

Examining 1354 TLDS (after excluding 1 TLD that had broken DNSSEC) on 2025-01-25

NSEC	**58**		**4.28%**
Classic	57	98.28%	
White Lies	0	0.00%	
Compact Denial	1	1.72%	
NSEC3	**1,296**		**95.72%**
Classic	1,296	100.00%	
White Lies	0	0.00%	
Compact Denial	0	0.00%	

NSEC3 Opt-Out: 1,106 of 1,296, or 85% of all NSEC3

Some Observations

- 1355 of 1445 TLDs signed (93.77%)
- 1 TLD had broken DNSSEC
- Only observed case of Online Signing with Minimally Covering NSEC is “GOV”, with Compact Denial of Existence on Cloudflare - see OARC42 Talk
- NSEC3 Opt-Out usage is very high.

Acknowledgements

For answering some of my questions and subsequent discussions ..

Petr van Dijk, PowerDNS

Puneet Sood, Google

Christian Elmerot, Cloudflare

Vladimir Cunat & Libor Peltan, CZ.NIC

Yorgos Thessalonikefs, NLNet Labs

David Blacka, Verisign

Roy Arends, ICANN

Jeffrey Damick, Amazon

A Survey of Authenticated Denial of Existence in DNSSEC

Shumon Huque

Thank You!

Questions or Comments?

To Sum Up ...

Authenticated Denial of Existence in DNSSEC is complex.

There are many varieties with differing features and tradeoffs (NSEC, NSEC3, Pre-Computed vs Online Signing, White Lies, Compact Denial of Existence, etc.).

A given auth DNS system may offer only specific modes.

Negative response synthesis is also varied in technique and adoption, & can pose issues with defective servers.

Online Signing with minimal NSEC is popular below the TLDs.

Zone enumeration defense is popular.

NSEC3 and Opt-Out is popular in the TLD space.