# On DNSSEC Negative Responses, Lies, and Zone Size Detection

Jonathan Demke[0000−0002−4162−2856] and Casey Deccio[0000−0003−0938−375X]

Brigham Young University, Provo, UT 84602, USA
{jpd0057,casey}@byu.edu
https://cs.byu.edu/

**Abstract.** The Domain Name System (DNS) Security Extensions (DNSSEC) introduced additional DNS records (`NSEC` or `NSEC3` records) into negative DNS responses, which records can prove there is no translation for a queried domain name. We introduce a novel technique to estimate the size of a DNS zone by analyzing the `NSEC3` records returned by only a small number of DNS queries issued. We survey the prevalence of the deployment of different variants of DNSSEC negative responses across a large set of DNSSEC-signed zones in the wild, and identify over 50% as applicable to our measurement technique. Of the applicable zones, we show that 99% are composed of fewer than 40 names.

**Keywords:** DNS · DNSSEC · Privacy.

## 1 Introduction

Since its inception over thirty years ago, the Domain Name System (DNS) [19,20] has included provisions for so-called negative responses, which indicate that there is no translation for a queried domain name. While the essential characteristic of a negative response has always been the lack of an *answer* (i.e., translation), the DNS Security Extensions (DNSSEC) [8, 19, 20, 23] introduced the requirement that additional DNS records (`NSEC` or `NSEC3` records) be included in a negative DNS response, which records can *prove* the non-translation of the domain name. A side effect of including these extra records is that additional information is revealed about a domain—such as names that *do* exist. While this side effect is innocuous to some, to others it can be undesirable. In an attempt to reduce or eliminate unwanted disclosure of information via DNSSEC negative responses, new approaches have been introduced into the DNSSEC ecosystem. However, each comes with its own caveats.

In this paper, we present a novel method for learning the *size* of a DNS zone—using DNSSEC negative responses—by issuing only a relatively small number of queries. From a standpoint of minimum information disclosure, even revealing the size of a zone might be a privacy concern to some entities. However, more generally it stands alone as a way to estimate zone size to learn more about how the DNS ecosystem is being utilized. We list the following as the major contributions of this paper:

- The presentation of a technique to estimate DNS zone size for `NSEC3`-signed zones;
- A measurement study on the use of different strategies of DNSSEC negative responses in the wild; and
- A survey of the sizes of various `NSEC3`-signed zones using the technique introduced in this paper.

As part of our study, we systematically issue queries to DNS servers authoritative for over two million DNSSEC-signed zones, eliciting negative DNS responses of various types. We find that over 50% of the zones we analyzed are signed with traditional `NSEC3`, and are thus candidates for zone size estimation using relatively few queries. We also observed that 99% of the `NSEC3` zones we analyzed have an estimated size of less than 40 names.

## 2 Background

The Domain Name System (DNS) [19, 20] protocol primarily consists of queries and responses. *Queries* are messages requesting the translation of a given *domain name* (i.e., `example.com`) and type (e.g., `A`, for IPv4 address). *Responses* are made of multiple DNS *records*, where a record is a mapping of domain name and type to some resource. The records in a DNS response collectively constitute either an *answer*, a *referral* to which server(s) might have the answer, or a definitive indication that there is no resource to which the name and type maps, i.e., there is no answer. A DNS *zone* is a group of DNS records with names under a common domain (i.e., suffix) and served from the same set of servers.

When there is no translation for a given name and type, the response includes no answer records, yielding an empty answer section. The `NSEC` record was introduced, with DNSSEC, to *prove* that for a given query a) the queried domain name doesn't exist or b) no record of the queried type exists at that name [8, 9]. An `NSEC` (next secure) record consists largely of two parts: 1) a pair of domain names that, using a defined canonical ordering, are in sequence; and 2) the list of types that exist for the first of the names in the `NSEC` record. If a queried name doesn't exist, the server returns the `NSEC` record that contains the names between which the queried name would fall, if it existed—the `NSEC` *covering*. If the queried name exists but the queried type does not, then the server returns the `NSEC` record corresponding to the name, and the list of types in the record prove that the queried type does not exist.

While `NSEC` records in a response provide a useful non-existence proof, their inclusion makes it possible for a server to divulge all existing domain names in a given zone through systematic querying. This exposure is a privacy concern for some organizations, but the introduction of `NSEC3` addressed this concern, in part [23]. With `NSEC3`, names within a DNS zone are hashed, and the ordered sequence of *hashes* that cover the hash of the queried name, are returned by a server, instead of the names that cover the queried name. Thus, the client receiving the response can prove non-existence of a given name, but doesn't immediately learn about any other names that do exist.

## 3 Previous Work

While NSEC3 effectively obfuscates the names from simple disclosure, research has shown that with a relatively small number of queries, a significant portion of zone contents can be enumerated using an offline dictionary "attack" [12, 24]. We complement this research to show that the *size* of a zone can also trivially be learned.

Further measures to protect DNS privacy by revealing less about a DNS zone involve servers sending minimal proofs—effectively "lying" about zone contents. Two major variants exist, one for NSEC3 records ("white lies") and one for NSEC records ("black lies"). The notion of NSEC3 white lies was introduced by Dan Kaminsky in his Phreebird DNSSEC software [18]. Upon receiving a query for a given domain name, $d$, rather than returning the NSEC3 record with the hashes corresponding to existing names that surround the hash of $d$, $h(d)$, the server dynamically creates an NSEC3 record with hashes $h(d) - 1$ and $h(d) + 1$. With the black lies approach—a term coined by Cloudflare—the server dynamically generates an NSEC record with 1) the name queried and 2) the next possible name in DNSSEC canonical ordering (i.e., `foo.com` and `\000.foo.com`) [16]. The result in both cases is an NSEC or NSEC3 proof that satisfies any validator without disclosing any existing names or hashes of existing names and does not disclose additional information. In the case of black lies, the response indicates that the name exists (even though it doesn't), but that the type does not.

Generating a dynamic response requires a server to have access to the private key(s) associated with the zone, so DNSSEC signatures (RRSIG records) can also be generated dynamically. This is in contrast to traditional static signing methods, in which RRSIG records can be created on a server, possibly even offline. This potentially creates concerns for zones served by third-party organizations [11, 22]; providing private keys to a third party allows them access to create arbitrary zone content. The NSEC5 mechanism was proposed to address this concern by providing a separate key to third parties, which was only good for providing a dynamic signature for an intermediate record that played a role in the proof [15]. Because this key cannot sign the records found in the zone proper, they cannot be used to manipulate. Despite the privacy advantages, NSEC5 has faced challenges with its standardization and adoption.

## 4 NSEC3 Zone Size Discovery

In this section we discuss the foundations and methodology for estimating the size of an NSEC3-signed zone.

It is well known that contents of zones signed with traditional NSEC (i.e., without black lies) can be trivially enumerated with a number of queries equal to the number of unique owner names in the zone. [21]. As a side effect, zone size—as measured by the number of unique owner names—is also discovered.

Zones signed with NSEC3 cannot be similarly enumerated. This is because the hashes returned in the NSEC3 records of one response cannot be used as

query names in subsequent responses, as they can in NSEC [23]. However, with traditional NSEC3 (i.e., no white lies), an interested party can accumulate a large number of NSEC3 records with repeated queries. With sufficient queries, the collection of records retrieved might approach the entire set of NSEC3 records for the zone. In that case, the investigator can not only carry out an offline dictionary "attack" [12, 24], but also learn the size of the zone.

Throughout the remainder of this paper, we refer to three types of queries used to elicit negative response, which we describe here:

- *q-nxdomain*: a type A query for a domain name within the zone, formed by pre-pending an arbitrary label of our choosing to the subject domain, e.g., foobar123.example.com, which domain name (presumably) does not exist.
- *q-nodata*: a type CNAME query for the domain name at the zone apex (i.e., the domain name corresponding to the zone itself), which record should also not exist (because a record of type CNAME cannot co-exist with the NS records also at the zone apex) [13].
- *q-nodata-type*: a query for an undefined type at the zone apex, which record should also not exist because the type has not been defined.

### 4.1 NSEC3 Distance

Like all DNS records, NSEC3 records have an owner name and record data. The first (left-most) label in the owner name is the Base32-encoded (using the "Extended Hex" alphabet [17]) value of the SHA1 hash of a domain name [23]. This label is 32 characters long, with each character having 32 possible values. The record data for an NSEC3 record includes, among other fields, the next field, which is the hash of another owner name in the zone—the next hash in the zone, in canonical ordering. The hash value in the next field can also be represented as a 32-byte string of Base32 characters. The maximum value of the Base32 representation of either is $H = 2^{160} = 32^{32}$.

The *distance*, $d(n)$, of an NSEC3 record, $n$, is the result of subtracting the next field's value, $n_{next}$, from the value of the first label of the owner name, $n_{owner}$. If the next hash value is greater than the hash value in the owner name, then the absolute difference is subtracted from the maximum hash space:

$$d(n) = \begin{cases} n_{owner} - n_{next} & \text{if } n_{owner} \geq n_{next} \\ H - |n_{next} - n_{owner}| & \text{otherwise} \end{cases} \tag{1}$$

For a given zone, $Z$, the sum of the distances of all the NSEC3 records must equal the total hash space, $H$:

$$\sum_{n \in Z} d(n) = H \tag{2}$$

### 4.2 NSEC3 Distance Distribution

To understand the distribution of distances within a zone, $Z$, we generated random names using the Natural Language Toolkit (NLTK) [10] to create 100 DNS

zones for each of the following zone sizes: $10^2$, $10^3$, $10^4$, $10^5$, and $10^6$. Each of the resulting 500 zones was signed with `NSEC3` using BIND's `dnssec-signzone` [1]. We then computed the distance for each `NSEC3` record and plotted the Cumulative Distribution Function (CDF) of all the distances, categorized by zone size, in Figure 1.

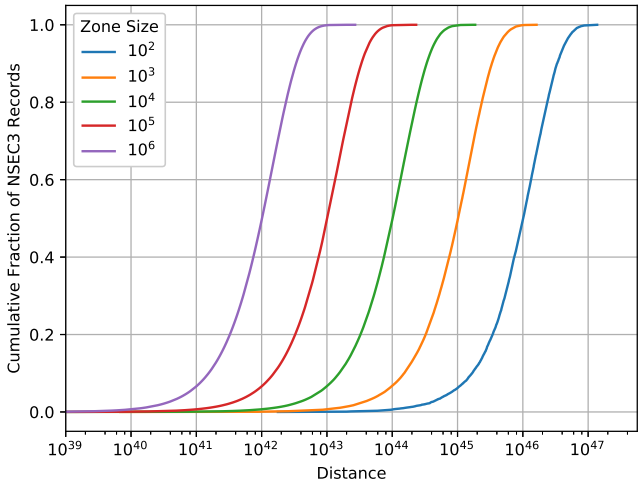


**Fig. 1.** The CDF of `NSEC3` distances for zones of various sizes.

The plots exhibit several noteworthy features. First, the plots of distances for each zone size are nearly identical, with each distribution being shifted from any other distribution according to the inverse proportionality of their respective zone sizes. For example, the median distance for the zone of size $10^2$, is 100 (i.e., $10^4/10^2$) times greater than the median distance for the zone of size $10^4$. Second, the CDF for each zone size increases logarithmically, rather than exhibiting a normal distribution. Thus, there is a much larger proportion of small `NSEC3` distances in each of the zones than large distances. It follows that for an `NSEC3`-signed zone, the majority of the hash space is covered by relatively few `NSEC3` records. Specifically, 90% of the hash space, $H$, is covered by only about 60% of the `NSEC3` records in a zone, and only 19% of the `NSEC3` records cover half of the hash space. Relatedly, the lower 50% of `NSEC3` distances for a given zone covers only 15% of the overall hash space, $H$.

The distribution of cumulative `NSEC3` distances to cumulative hash space, for the 500 zones we created, is shown in Figure 2. Notably, Figure 2 plots the same `NSEC3` distance data as Figure 1, consolidating the distance data from all the zones. Because the distribution of `NSEC3` across the hash space is the same for any zone size, the resulting plot is a single, unified line.

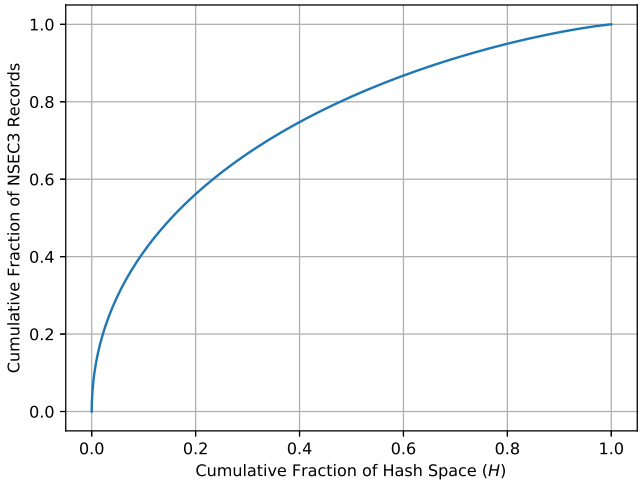


**Fig. 2.** The CDF of `NSEC3` distances compared to cumulative percentage of hash space.

The hashes of query names, however, are distributed uniformly across the hash space. We confirmed this by generating 100,000 unique query names within a domain and analyzing the distribution of the resulting `NSEC3` hashes, which were computed using BIND's `nsec3hash` utility [1]. The hash space, $H$, was divided up into 1,024 equal-sized bins, and the number of `NSEC3` hashes that fell in each bin was graphed as a CDF, shown in Figure 3. The number of `NSEC3` hashes per bin were normally distributed with a median value of 98, which is the expected value for 100,000 queries, i.e., $100000/1024 = 98$.

The apparent disparity between the uniform distribution of hashes and the exponential distribution of the distances between them is actually an example of a Poisson process. The `NSEC3` hashes represent "arrival times" across the hash space, which are uniformly distributed according to constant *intensity* (or *arrival rate*) $\lambda$, which is a function of the size of the zone. The `NSEC3` distances represent the inter-arrival times and are distributed according to $\texttt{Exp}(\lambda)$ [14].

Let $z = |Z|$ denote the actual size of DNS zone $Z$, and let $z'$ represent the estimate of $z$, derived from `NSEC3` distances. If the distances of all `NSEC3` records were somewhat uniform, then calculating $z'$ would be as simple as calculating the average distance of the collection of `NSEC3` records, $N$, returned in negative responses to *q-nxdomain* queries and dividing $H$ by that average:

$$z' = \frac{H}{\left(\frac{\sum_{n \in N} d(n)}{|N|}\right)} \tag{3}$$

However, the fact that the distribution of `NSEC3` distances—for a zone of any size—follows an exponential distribution across the hash space, while the distri-

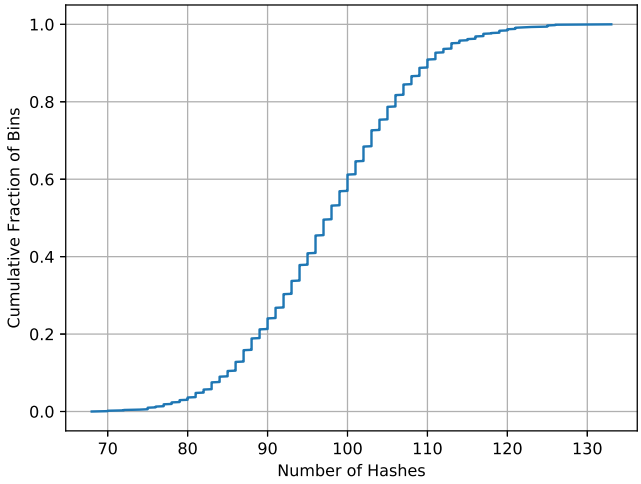


**Fig. 3.** The distribution of NSEC3 hashes resulting from random query names, graphed as a CDF representing the number of hashes per 1,024th part of the hash space, $H$.

bution of NSEC3 hashes are uniformly distributed across the hash space, means that not all queries are equal. That is, the NSEC3 hash corresponding to an arbitrary *q-nxdomain* query is more likely to be covered by an NSEC3 record with a large distance, but that distance is less representative of the zone's NSEC3 records because of the relatively large percentage of NSEC3 records having a smaller distance. Thus, if all NSEC3 distances in $N$ were weighted equally (i.e., following Equation 3), then the resulting average would be too high, resulting in a proportionally too-low value for $z'$.

A more accurate approach to estimating the size of a zone using the collection of NSEC3 records, $N$, returned in negative responses to *q-nxdomain* queries, is to weight each NSEC3 record according to its statistical representation of the hash space. If the NSEC3 records in $N$ are divided into $q$ quantiles, according to their distance, $N = N_1 + N_2 + \ldots + N_q$, then the distance for all records in $N_i$ are weighted using the fraction of the hash space that that $i$th quantile represents. The weights for $q = 10$ (i.e., decile or 10th percentiles) were derived from the distance distribution of the NSEC3 records from the 100 zones of size $10^6$ that we created and are shown in Table 1. These weights correspond to the difference in cumulative hash space, $x$, for consecutive quantile values of NSEC3 distances, i.e., $y_1 = \frac{i-1}{q}$ and $y_2 = \frac{i}{q}$. The resulting formula to approximate zone size, letting $w_i$ correspond to the weight for quantile $i$, is the following:

$$z' = \frac{H}{\left(\sum_{1 \le i \le q} \frac{w_i \sum_{n \in N_i} d(n)}{|N_i|}\right)} \tag{4}$$

**Table 1.** Distance weights for zone size detection using decile divisions (i.e., $q = 10$).

| Decile ($i$) | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| Weight ($w_i$) | 0.41 | 0.15 | 0.10 | 0.08 | 0.07 | .05 | .05 | 0.04 | 0.03 | 0.02 |

The result of this weighted approach is that the NSEC3 records with larger distances—which are more likely to cover an arbitrary *q-nxdomain* query but are less representative of the zone's NSEC3 distances—contribute less to the average than NSEC3 with small distances—which are less likely to cover an arbitrary *q-nxdomain* query and are more representative of the zone's NSEC3 distances.

### 4.3 Validation

To test the validity of our zone size detection methodology, we issued 1,000 trials, each consisting of 18 *q-nxdomain* queries, for each of the zones we created. The zones were served locally on a BIND DNS server. For each trial, the 18 queries yielded a total of 20 NSEC3 records, which comprised $N$; in addition to the 18 NSEC3 records covering the unique names queried, every response included the NSEC3 record that covers the wildcard record and the NSEC3 record matching the zone name [23].

First, we investigated the accuracy of our methodology using different quantile ($q$) values. Specifically, we evaluated the 1,000 trials against the zone of size 10,000 using quantile values of 20, 10, and 5. We measured accuracy in terms of percentage of error based on the actual zone size, i.e.,

$$e = \frac{z' - z}{z} \tag{5}$$

Thus, values of $e$ closer to 0 indicate higher accuracy of zone size prediction, $e < 0$ indicates a low guess ($z' < z$), and $e > 0$ indicates a high guess ($z' > z$). The results are shown in Figure 4 as a CDF.

Weighting the NSEC3 distances, by any quartile value (Equation 4) significantly improved the accuracy from the zone size estimates based on unweighted distance averages (Equation 3). Even the highest zone size estimates calculated using unweighted averages were lower than the actual zone size, with the median error being about 48% low. In contrast, for about 60% of the trials (between the 30 and 90 percentiles) for $q = 10$ and $q = 20$, $z'$ was within 15% of $z$. And for about 30% of the trials (between the 50 and 80 percentiles), the $z'$ was within 7% of $z$. Because the error for $q = 10$ and $q = 20$ were comparable, and $q = 10$ requires fewer queries to have at least one NSEC3 record in every quantile, we use $q = 10$ for the remainder of our experiments.

We next tested the accuracy of zone size prediction against zones of different sizes, the results of which are shown in Table 2 and Figure 5. Consistent among the zones of all sizes was that $z'$ was low more often than not, with the median values of $e$ ranging between -6% and -16%. For zones smaller than 100,000, 75%

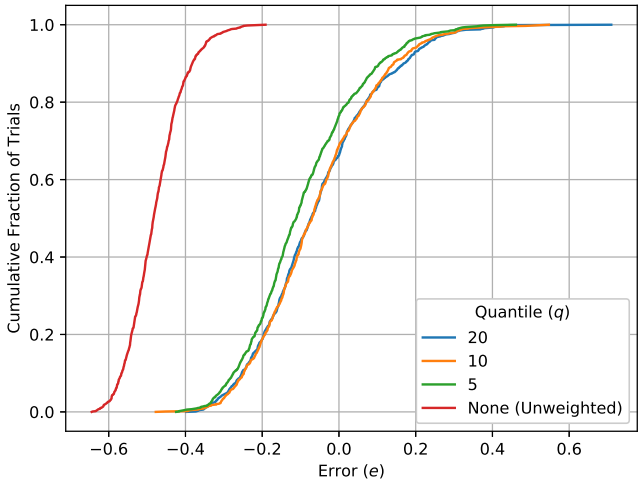


**Fig. 4.** Error ($e$) for size prediction of a DNS zone of size $10^6$ for various values of $q$.

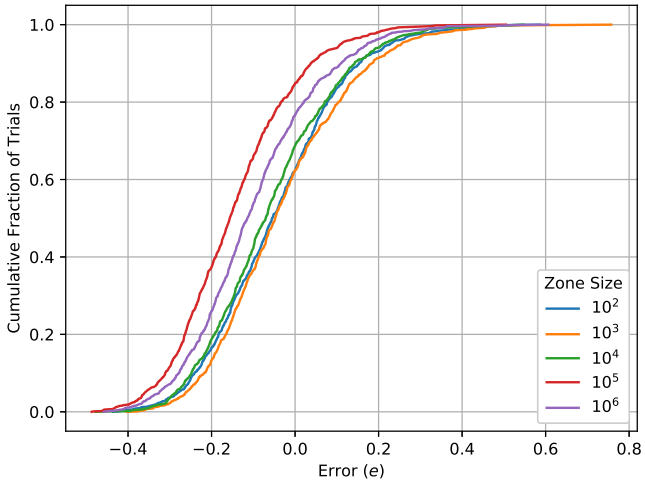


**Fig. 5.** Error ($e$) for zone size prediction of DNS zones of various sizes using $q = 10$.

**Table 2.** Statistics for zone size prediction of DNS zones of various sizes using $q = 10$.

| Zone size | $10^2$ | $10^3$ | $10^4$ | $10^5$ | $10^6$ |
|---|---|---|---|---|---|
| Median error | -0.06 | -0.05 | -0.07 | -0.16 | -0.11 |
| Fraction of trials $-0.20 \leq e \leq 0.20$ | 0.77 | 0.78 | 0.76 | 0.61 | 0.70 |

or more of the trials had error values that were within 20% of the size of the zone.

## 5 DNS Zone Size Measurement Study

To perform our DNS zone size measurement on deployed DNS zones, we analyzed 2,182,987 DNSSEC-signed zones to determine the strategy they employ for negative responses. This would allow us to identify our candidate DNS zones. The list of zones consisted of DNSSEC-signed second-level domains extracted from the zone files for 821 top-level domains (TLDs). The TLD zone files themselves were obtained from the following sources: Verisign's Zone File Access [6]; the Centralized Zone Data Service (CZDS) [2]; the Public Interest Registry (PIR) [5]; the Internet Foundation in Sweden (IIS) [4]; and Domains Index [3], from which we acquired domains under `gov`. DNSSEC-signed domains were identified as those with at least one delegation signer (`DS`) record in the TLD zone file. The breakdown of the domains and their TLD are shown in Table 3. Nearly 80% of

**Table 3.** Breakdown of domains analyzed, both by TLD and by detected negative response type.

| TLD | Zones Analyzed | Traditional `NSEC` | Traditional `NSEC3` | White Lies `NSEC3` | Black Lies `NSEC` | Unclassified |
|---|---|---|---|---|---|---|
| `com` | 911,576 (42%) | 112,168 | 725,521 | 18,879 | 36,501 | 17,823 |
| `se` | 802,198 (37%) | 77,549 | 147,294 | 539,178 | 408 | 37,072 |
| `net` | 127,545 (6%) | 14,390 | 103,136 | 2,762 | 5,089 | 1,920 |
| `nu` | 118,158 (5%) | 9,508 | 33,801 | 66,690 | 74 | 7,623 |
| `org` | 95,319 (4%) | 9,252 | 79,557 | 2,214 | 2,964 | 1,076 |
| `app` | 33,254 (2%) | 492 | 7,223 | 25,232 | 219 | 33 |
| Other | 94,937 (4%) | 17,686 | 70,687 | 2,136 | 2,804 | 1,099 |
| Total | 2,182,987 (100%) | 241,045 (11%) | 1,167,219 (53%) | 657,091 (30%) | 48,059 (2%) | 66,646 (3%) |

the zones analyzed were under the `com` and `se` TLDs. This was because of the significant presence of `DS` records in those domains.

### 5.1 Zone Analysis

For each zone in our data set, we identified the authoritative servers using DNS lookups for the `NS` (name server) records and the corresponding `A` and `AAAA` (IPv4 and IPv6 address) records. Having the set of IP addresses for servers authoritative for the domains, we issued three queries to every authoritative server: a *q-nxdomain* query, a *q-nodata* query, and a *q-nodata-type* query. The three queries were intended to elicit different types of negative response behavior, including any of the following:

- `NXDOMAIN`: a response indicating that the name queried name doesn't exist.
- wildcard: a response synthesized from a wildcard, with `NSEC` or `NSEC3` records to indicate that the queried name didn't exist (as specified by DNSSEC [23]).
- `NODATA`: a response indicating that the name exists, but with no records corresponding to the type queried [7].

The expected response for *q-nxdomain* was either an `NXDOMAIN` or wildcard response, and the expected response for *q-nodata* and *q-nodata-type* was `NODATA`. Under DNSSEC requirements, all such responses would include `NSEC` or `NSEC3` records. Table 3 shows the breakdown of response strategies observed by authoritative servers: traditional `NSEC`, traditional `NSEC3`, white lies with `NSEC3`, and black lies with `NSEC`. If at least one of the query responses matched a given negative response strategy, then the zone was included in the count for that strategy. We note that for a very small (less than 1%) percentage of the zones analyzed, we observed several different negative response behaviors, such that they are represented in multiple categories. For example, for some zones, white lies was used in response to our *q-nxdomain*, but `NSEC` records were returned in response to the *q-nodata-type*. Also, for 3% of the DNS zones we analyzed, none of our queries resulted in `NSEC` or `NSEC3` records, so their negative response strategy remained unclassified.

The responses for the unclassified zones fell into several categories. Some of the *q-nxdomain* queries yielded non-wildcard positive responses (i.e., indicating that the record existed), the result of server-side record synthesis with online-signing. This method is employed by organizations in an effort to not even disclose the fact that the response is a wildcard—which would otherwise be apparent. Some responses lacked `NSEC` or `NSEC3` records due to misconfiguration. For example, a `DS` record existed, but the zone was actually not DNSSEC-signed, or the response had response code `SERVFAIL`.

We observed nearly one-third of the zones employed white lies, while just over half used traditional `NSEC3`. About 11% of zones were signed with traditional `NSEC`, while about 2% of zones used black lies. The combined presence of white lies and black lies implied that a minimum of 32% of the zones we analyzed employed a online signing.

### 5.2 Detecting Zone Size in the Wild

We tested our zone detection methodology in the wild by issuing 20 queries to each of the zones in our dataset that were signed with plain `NSEC3`, i.e., without white lies. The results of this measurement are shown in Figure 6. We found that 85% of the zones we probed were so small that even with only 20 queries, we received fewer than 10 unique `NSEC3` records, which is the minimum size of $N$ necessary to apply our methodology. The fact that `NSEC3` records were being returned multiple times with these zones was evidence that the zone was small, and was—quite likely—being completely enumerated with our small number of queries. Thus, for $|N| < 10$, we simply use $z' = |N|$ as our zone size estimate. For the zones we measured, 99% were smaller than 40, but the top 1% reached up to nearly four million.

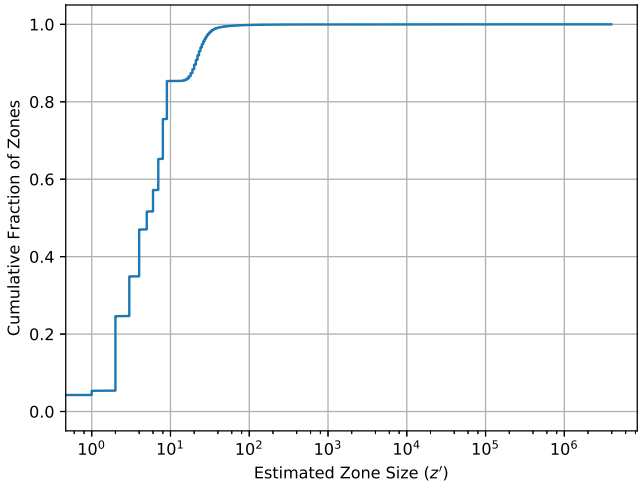


**Fig. 6.** Estimated zone sizes ($z'$) for the `NSEC3`-signed zones in our data set (Table 3).

## 6 Conclusion

In this paper, we have presented methodology for learning the size of a DNS zone by issuing relatively few DNS queries. We demonstrated the accuracy of our technique in a lab environment and showed that in approximately 75% of cases, the methodology would yield a an estimate that is within 20% of the actual zone size, with only 18 queries. We deployed this methodology on over one million `NSEC3` zones in our data set and learned that most of these zones are small, with 85% having fewer than 10 domain names.

As part of our study, we measured some of the DNSSEC negative response behaviors currently deployed. We learned that the most popular negative response strategy deployed in our data set is traditional `NSEC3`, which is used by 53% of zones, and makes them candidates for DNS zone size estimation, using our methodology. Privacy-preserving strategies such as `NSEC3` with white lies and `NSEC` with black lies are also gaining some traction with 30% and 2% deployment, respectively.

The techniques presented in this paper serve as a general purpose tool to better understand the DNS ecosystem, in terms of the size of deployed DNS zones, specifically those signed with `NSEC3`. It also provides a new insight into information disclosure, regardless of how innocuous the revealing of the size of DNS zone might be to an organization. This knowledge can only benefit and empower the designers, maintainers, and users of the Internet.

## References


1. BIND open source DNS server, https://www.isc.org/downloads/bind/
2. Centralized zone data service, https://czds.icann.org/
3. Domains index, https://domains-index.com/
4. The internet foundation in sweden, https://www.iis.se/
5. Public interest registry, https://pir.org/
6. Verisign, https://www.verisign.com/
7. Andrews, M.: RFC 2308: Negative caching of DNS queries (DNS NCACHE) (March 1998)
8. Arends, R., Austein, R., Larson, M., Massey, D., Rose, S.: RFC 4033: DNS security introduction and requirements (March 2005)
9. Arends, R., Austein, R., Larson, M., Massey, D., Rose, S.: RFC 4034: Resource records for the DNS security extensions (March 2005)
10. Bird, S., Loper, E., Klein, E.: Natural Language Processing with Python. OReilly Media Inc. (2009)
11. Deccio, C., Chen, C.C., Mohapatra, P., Sedayao, J., Kant, K.: Quality of name resolution in the domain name system. In: 2009 17th IEEE International Conference on Network Protocols (October 2009)
12. DNSCurve: DNSCurve: Usable security for DNS, http://dnscurve.org/nsec3walker.html
13. Elz, R., Bush, R.: RFC 2181: Clarifications to the DNS specification (July 1997)
14. Gardiner, C.: Stochastic Methods: A Handbook for the Natural and Social Sciences. Springer (2009)
15. Goldberg, S., Naor, M., Papadopoulos, D., Reyzin, L., Vasant, S., Ziv, A.: NSEC5: Provably preventing DNSSEC zone enumeration. In: NDSS'15 (February 2015)
16. Grant, D.: Economical with the truth: Making DNSSEC answers cheap, https://blog.cloudflare.com/black-lies/
17. Josefsson, S.: RFC 4648: The base16, base32, and base64 data encodings (October 2006)
18. Kaminsky, D.: Phreebird, https://dankaminsky.com/phreebird/
19. Mockapetris, P.: RFC 1034: Domain names - concepts and facilities (November 1987)
20. Mockapetris, P.: RFC 1035: Domain names - implementation and specification (November 1987)
21. Osterweil, E., Ryan, M., Massey, D., Zhang, L.: Quantifying the operational status of the dnssec deployment. In: Proceedings of the 6th ACM/USENIX Internet Measurement Conference (IMC'08) (October 2008)
22. Ramasubramanian, V., Sirer, E.G.: Perils of transitive trust in the domain name system. In: IMC '05 Proceedings of the 5th ACM SIGCOMM conference on Internet measurement (October 2015)
23. Sisson, G., Arends, R., Blacka, D.: RFC 5155: DNS security (DNSSEC) hashed authenticated denial of existence (March 2008)
24. Wander, M., Schwittmann, L., Boelmann, C., Weis, T.: GPU-based NSEC3 hash breaking. In: 2014 IEEE 13th International Symposium on Network Computing and Applications. IEEE (August 2014)