SAND2011-6346C

Maintenance, Mishap, and Mending in DNSSEC Deployment

No Author Given

No Institute Given

Abstract. The Domain Name System Security Extensions (DNSSEC) add an element of authentication to the DNS, which is a foundational component of today's Internet. However, maintenance of a DNSSEC deployment is more complex than that of its insecure counterpart. In this paper we identify some of the specific misconfigurations impacting DNSSEC deployments, analyze their pervasiveness from an extended survey of production DNS, and assess maintenance and corrective behaviors. Over half the zones we analyzed have been affected by misconfiguration. We also observed a significant presence of repeat occurrence and average correction times of up to two weeks. We summarize our findings and suggest insights towards improving the quality of DNSSEC deployment.

Keywords: DNS, DNSSEC, Misconfiguration

1 Introduction

The Domain Name System (DNS) [14, 15] is a distributed database for looking up data based on domain name and query type and is one of the foundational components of the Internet. The most common use is mapping domain names to Internet addresses.

The DNS Security Extensions (DNSSEC) [5–7] were introduced to protect the integrity of DNS responses. DNSSEC allows DNS administrators to cryptographically sign and validate DNS data. The number of DNSSEC-signed zones has increased significantly in the last year, including the DNS root zone and a large number of top-level domains (TLDs) [2, 3, 10]. However, in order to achieve its security benefits, DNSSEC adds non-trivial complexity to the DNS. This increases the chances of a DNS outage if not properly deployed or maintained. The effects of misconfiguration have been felt at various levels in the DNS hierarchy, including TLDs, and even the root zone. An understanding of DNSSEC components, their relationship, and the protocol itself are essential for proper deployment.

In this paper we analyze DNSSEC deployment over the past year to answer the following questions:

- What DNSSEC maintenance practices are being employed?
- What is the prevalence of misconfiguration among DNSSEC deployments?
- How are operators addressing broken DNSSEC deployments?

We base our analysis on a survey of a sample of DNSSEC-signed zones polled over an extended period of time. We use the results of our analysis to suggest tool functionality to improve the quality of DNSSEC deployment.

The remainder of our paper is organized as follows. In Section 2 we provide a brief review of DNS and DNSSEC, and in Section 3 we outline challenges associated with DNSSEC. In Section 4 we describe our survey of DNSSEC deployment and analyze the results. We refer to previous work in Section 5 and conclude in Section 6.

2 DNS Background

In the DNS [14, 15] a *resolver* queries *authoritative* servers to receive answers. It learns authoritative servers for a DNS *zone* by starting at the *root* zone and following referrals downward in delegated DNS namespace until it receives an authoritative response. Queries include a name and type, and answers are comprised of *resource records* (RRs), which have a name, type, and record data. Resource records are grouped by name and type into *RRsets*.

DNSSEC [5–7] adds authentication to the DNS. RRsets are signed on a per-zone basis, and each signature is contained in an `RRSIG` RR. Authoritative servers return `RRSIG`s with any RRsets they cover. A zone's public keys are published in the zone's `DNSKEY` RRset. Having an `RRSIG` and the corresponding `DNSKEY` a validating resolver can verify the integrity of the RRset it covers.

DNSSEC scales by establishing a *chain of trust* upwards through the namespace hierarchy, and anchoring with the `DNSKEY` of a common ancestor zone, typically the root. The link between zones is accomplished by the introduction of `DS` (*delegation signer*) RRs in the parent zone. A `DS` includes the cryptographic digest of a `DNSKEY` in the child zone of the same name. When the `DNSKEY` corresponding to an authenticated `DS` or trust anchor is used to the sign the the zone's `DNSKEY` RRset, it becomes a *secure entry point* (SEP), and all `DNSKEY`s in the RRset are authenticated. A common setup is for a zone to sign only its `DNSKEY` RRset with the SEP key (a *key signing key* or KSK) and sign other zone data with a second key (a *zone signing key* or ZSK).

Authenticated denial of existence is accomplished using `NSEC` (*next-secure*) RRs, which are provided in a response to show a validator where the non-existent RRset would appear (in a canonical ordering of the zone) if it did exist. *Hashed authenticated denial of existence* using `NSEC3` RRs is a newer protocol introduced to address challenges inherent in the use of `NSEC` [13].

3 DNSSEC Challenges

DNSSEC carries additional maintenance considerations, and negligence or misconfiguration may result in validation failures. We briefly discuss DNSSEC maintenance and misconfiguration in this section.

A zone signed with DNSSEC requires more careful maintenance than an unsigned zone. RRSIGs have a limited lifetime, so the RRsets they cover must be periodically re-signed to replace RRSIGs that would otherwise go stale.

While DNSKEYs technically do not expire, it is recommended that they periodically be replaced to prevent prolonged exposure. Such replacement is called a *key rollover*. Best current practices for key rollovers are documented in RFC 4641 [12]. Non-SEP DNSKEYs can be rolled without involving third-parties and are thus self-contained. However, when a SEP DNSKEY is rolled, the parent zone must be involved to handle the change in DS RRs. Likewise, a validator must be engaged when a configured trust anchor is rolled [22].

Misconfiguration of a DNSSEC deployment results in a break in the chain of trust and a bogus validation outcome for the RRsets involved. Such a break also invalidates any dependent RRsets, including those in descendant zones. We enumerate six specific misconfigurations, to which we refer in the remainder of the document.

DS Mismatch If DS RRs are present in a parent zone but none correspond to any self-signing DNSKEYs in the child zone, the result is a bogus delegation, and RRsets in the child zone and below are deemed bogus.

DNSKEY Missing If a DNSKEY referenced in an RRSIG or DS is necessary to complete a chain of trust, but not included in the DNSKEY RRset, then validation fails.

NSEC Missing The lack of NSEC RRs does not necessarily present a major problem for the owning zone itself. However, it is more consequential for unsigned child zones not securely linked to their parent. NSEC RRs are required in a referral or in response to a DS query for the insecure delegation to show that no DS RRs exist. Without such the chain of trust is broken.

RRSIG Missing If an authoritative server does not provide the RRSIGs necessary to complete a chain of trust for a given RRset, then the chain of trust is broken, and the result is a bogus validation.

RRSIG Bogus The signature in the record data of an RRSIG must validate against the RRset it covers, or it is invalid.

RRSIG Dates If an RRSIG is allowed to expire, or is published before its inception date, then it fails to validate.

4 DNSSEC Deployment Survey and Analysis

We describe in this section a survey of DNSSEC deployment and analyze results.

Our survey consisted of periodic polling of production DNS zones signed with DNSSEC during a timespan of over one year—June 2010 to July 2011. We analyzed each signed zone several times daily, querying each authoritative

server to elicit various DNSSEC-related responses. Our zones came from three sources: hostnames extracted from URLs indexed by the Open Directory Project (ODP) [17]; names queried to recursive resolvers at the SC08 conference [21]; and names submitted by third parties to a Web-based analysis tool.

We identified production signed zones in our data set by considering only zones indicating their public intent to be validated by resolvers—those with an authentication chain to the root zone trust anchor (after the July 2010 signing of the root [2]) or to the trust anchor at ISC's *DNSSEC Look-aside Validation* (DLV) service [11]. DLV [24] was introduced to allow an arbitrary zone to be securely linked to a zone other than its hierarchical parent for trust anchor scalability prior to the root signing.

To further avoid zones set up for non-production testing we excluded zones containing the names "test", "bogus", "bad", and "fail", and those that were subdomains of known DNSSEC test namespaces (e.g., *dnsops.gov* and *dnsops.biz*, of the Secure Naming Infrastructure Pilot [16]). The total number of production signed zones analyzed was 2,242, though the total number analyzed during any given polling period varied as new zones were added or as monitored zones entered or left production DNSSEC.

The breakdown of analyzed zones by TLD is shown in Figure 1. For signed zones under the *gov* TLD, which made up the largest contingency of those analyzed, 40% of zones experienced some type of misconfiguration. For nearly all TLDs shown, at least 30% experienced some type of misconfiguration.

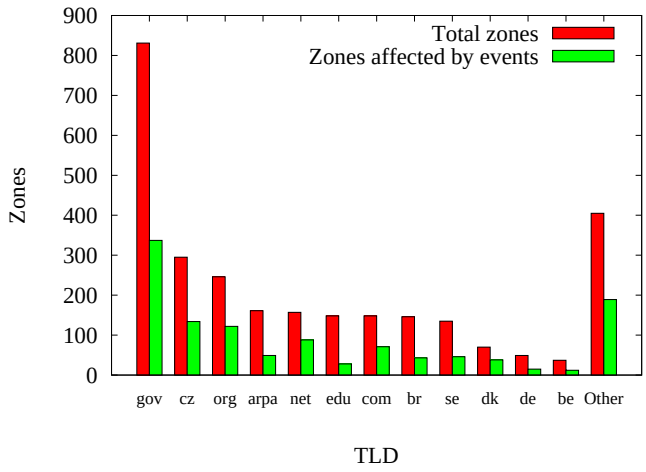

Fig. 1. TLD breakdown of production signed zones analyzed, and zones that exhibited misconfiguration during our survey.

4.1 DNSSEC Maintenance Observations

We first report on maintenance practices observed during our survey. We present in Figure 2 the average lifetime of `RRSIG`s covering the `DNSKEY` and `SOA` (*start of authority*) RRsets, made by the KSK and ZSK respectively, as a cumulative distribution function (CDF). This gives us an idea of the frequency at which administrators are required to re-sign their zone data. The distributions for `RRSIG` lifetimes of each key type mirror each other for all but about 2% of cases, in which the lifetime of `RRSIG`s made by a KSK approach 400 days, contrasted with ZSK `RRSIG`s with 30-day lifetimes. Other than this gap, the upper bound for accessibility of the corresponding private keys of both types is roughly the same. About 44% of zones had an average `RRSIG` lifetime of 30 days—the large vertical jump in the CDF—and 25% had an average `RRSIG` lifetime of 10 days or less. Nearly 5% of zones maintained `RRSIG`s with lifetimes over a year.

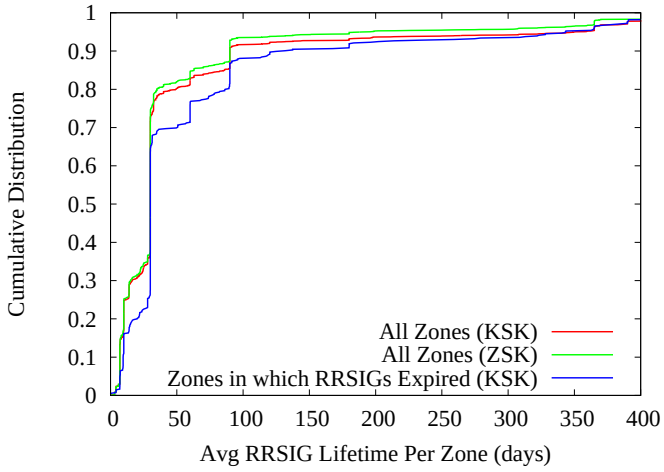

Fig. 2. CDF for average lifetimes of `RRSIG`s covering `DNSKEY` and `SOA` RRsets of production signed zones, made respectively by the zones' KSKs and ZSKs.

The average lifetime of `RRSIG`s (made by KSKs) for the subset of zones that allowed their `RRSIG`s to expire is also represented in Figure 2. This shows that zones maintaining `RRSIG`s with longer lifetimes and lower re-signing frequencies have a larger showing of `RRSIG` expiration. While 25% of all zones must be re-signed at least every 10 days, only about 15% of the zones having expirations have similar requirements. Roughly 32% of the zones experiencing expirations have `RRSIG` lifetimes greater than 30 days, although only about 24% of all zones have `RRSIG` lifetimes of similar magnitude.

During our survey we observed key rollovers to analyze the lifetimes of ZSKs and KSKs. Only zones for which we observed two or more rollovers could we

accurately determine the average lifetime of keys of either role. Although, for zones that experienced a single rollover we estimated lifetime using the maximum of the time monitoring the zone prior to the rollover and the time from the rollover to the completion of our survey. A breakdown of the rollovers we observed per zone is shown in Table 1. We can't judge the significance of the 90% of zones

	Number of Rollovers		
Key role	0	1	2+
ZSK	1,049 (37%)	309 (11%)	1470 (52%)
KSK	72%	492 (17%)	271 (10%)

Table 1. Key rollovers per zone observed during our survey.

that performed fewer than two rollovers during our survey given that some best practice documents recommend rollovers every 1 – 2 years [8]. However, we might sense some hesitancy to implement the potentially troublesome KSK rollover in the 72% of zones for which no rollover was detected. While the ZSK rollovers are less complex and require no coordination with the parent zone, we find it interesting that 11% performed only a single ZSK rollover, and 37% performed no ZSK rollovers during our survey.

The CDF of the average ZSK and KSK lifetimes for zones with one or more respective rollovers is shown in Figure 3. Half of the sample zones that performed two KSK rollovers did so on average over 75 days apart, and 55% of zones that performed two ZSK rollovers did so in 30 days or less, on average.

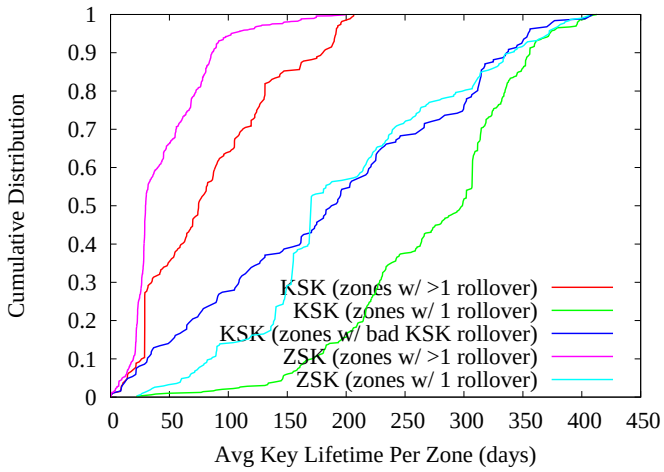

Fig. 3. CDF for average lifetimes of ZSKs and KSKs for zones having performed one or more rollovers during our survey.

Figure 3 also plots the average KSK lifetime for zones that experienced bad KSK rollovers (i.e., resulting in DS mismatches) during our survey. The distribution of KSK lifetimes for these zones is fairly even, indicating there is no significant correlation between KSK rollover frequency and DS mismatch, based on our observation. Half of the zones that experienced bad KSK rollovers averaged a KSK lifetime of more than 200 days, while half of zones rolled their KSKs within 200 days.

4.2 DNSSEC Misconfiguration Pervasiveness

Throughout our survey we identified the DNSSEC misconfigurations described in Section 3 to determine their pervasiveness in production. Our analysis was based on authentication of the SOA RRset for each zone. If we observed a misconfiguration in two or more consecutive polling intervals (i.e., at least four hours), then we classified it as a single event. Misconfigurations with only a single data point were not included in our analysis.

We identified 5,941 unique instances of misconfiguration in 1,172 zones over the duration of our survey. It is possible that misconfigurations of different natures contributed to a single outage, such as expired RRSIGs on a DNSKEY already suffering from a DS mismatch. The breakdown of misconfigurations observed is shown in Figure 4, also summarized in Table 2. For each misconfiguration type, the instances are subdivided based on whether they affected all authoritative servers (complete), a subset of servers only (partial, i.e., due to server inconsistency), or were resolved iteratively (incremental). Incremental misconfiguration means that over several polling periods the misconfiguration was present on all servers, then a subset, before finally being resolved.

Misconfiguration type	Total instances	TLDs Affected	Complete or Incremental	Avg duration (days)	Repetition rate
DS Mismatch	392	0	392 (100%)	12	22%
DNSKEY Missing	1,250	1	98 (8%)	6.8	40%
NSEC Missing	449	3	21 (5%)	14	39%
RRSIG Missing	1,819	5	0 (0%)	7.0	51%
RRSIG Bogus	272	0	208 (76%)	5.7	42%
RRSIG Dates	1,759	1	(67%)	7.5	46%

Table 2. A summary of DNSSEC misconfiguration instances observed in our survey.

The largest contributor to certain DNSSEC validation failure is RRSIGs with invalid dates. This typically refers to expired RRSIGs, although we identified several misconfigured zones that repeatedly signed their zones with future inception dates, never quite allowing their RRSIGs to reach the validity period. RRSIG expiration requires the least administrator intervention to achieve; it occurs with the negligence to re-sign the zone. In the 572 partial and 117 incremental occurrences, it is likely that the zone was re-signed (before RRSIG expiration, in the

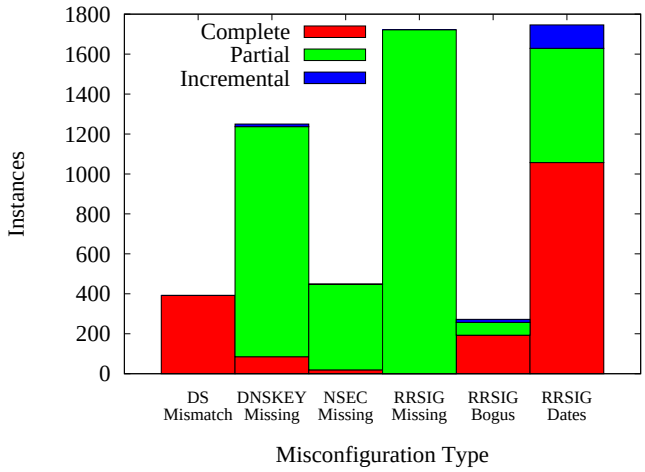

Fig. 4. Instances of DNSSEC misconfiguration observed, grouped by type and whether zones were completely affected, partially affected, or corrected incrementally.

case of partial misconfiguration), yet some of the authoritative servers did not transfer the most recent version of the fresh zone. The next largest contingency of complete misconfiguration is attributed to `DS` mismatch, of which there were 392 instances.

Missing `RRSIG`s are the leading contributor to possible failure due to partial misconfiguration. We attribute this to zones that were signed but for which one or more authoritative servers do not support DNSSEC and therefore do not return `RRSIG`s appropriately. We observed several zones with servers that "flapped" between returning `RRSIG`s and not.

4.3 Misconfiguration Resolution

We analyze several aspects of the misconfiguration instances we observed to understand the resolution path taken by administrators to correct problems. Two paths can be taken to appease resolvers validating misconfigured zones: correct the configuration; or remove the `DS` RRs for the zone, creating an insecure delegation. Misconfiguration instances are categorized by corrective action in Figure 5. Unresolved events persisted beyond the lifetime of our survey, so their resolution is unknown. The misconfiguration that most frequently saw anchor removal relative to proper correction was `DS` mismatches, at 20%. It also was second to invalid `RRSIG` dates in events yet unresolved.

We next examine the duration of each misconfiguration instance to understand the responsiveness of DNS administrators in identifying and taking corrective action for a DNSSEC-related problem, whether by removing its anchor or properly correcting the problem. Figure 6 plots the lifetime of each event as a cumulative distribution function (CDF), and the averages are shown in Table 2.

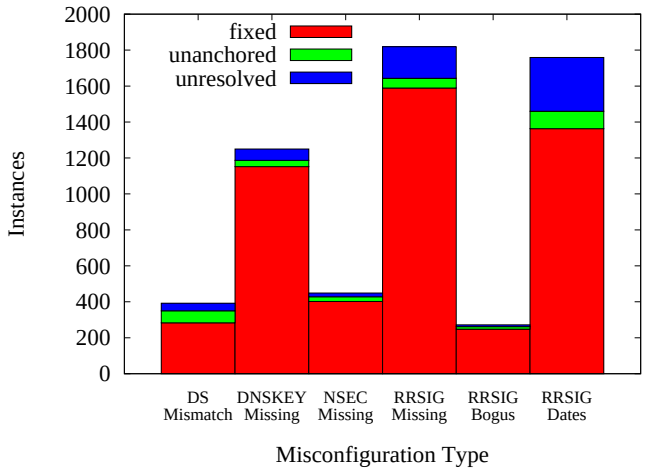

Fig. 5. Corrective action taken for resolution of DNSSEC misconfiguration instances.

Note that we exclude from our event duration results from events which were unresolved at the conclusion of our survey.

On average, events involving `RRSIG`s with bogus signatures were corrected within the least amount of time, followed by missing `RRSIG`s and `RRSIG`s with invalid dates. This is not surprising since the remedy for each of these is simply re-signing the of the zone which replaces the affected `RRSIG`s. At least half of each of these misconfigurations were corrected within two days.

Events involving missing `NSEC` RRs averaged the largest correction time at two weeks, likely because the misconfiguration is more subtle in most cases. `DS` mismatches took 12 days to correct on average, and one in four took over a week to correct.

4.4 Repeated Misconfiguration

Figure 7 shows the number of events per zone, classified by type, and Table 2 includes the repetition rate for different misconfigurations.

`DS` mismatches were repeated by one in five affected zones, a significant proportion, but the smallest in comparison to other misconfigurations. The relative infrequency could be because KSK rollovers generally happen less frequently than `RRSIG` renewals, so there is less opportunity for error. Missing `RRSIG`s and `RRSIG`s with bad dates account for the highest occurrence of repeat offense, averaging 46% and 51%. Almost 30% of zones affected by missing `RRSIG`s experienced this misconfiguration more than three times.

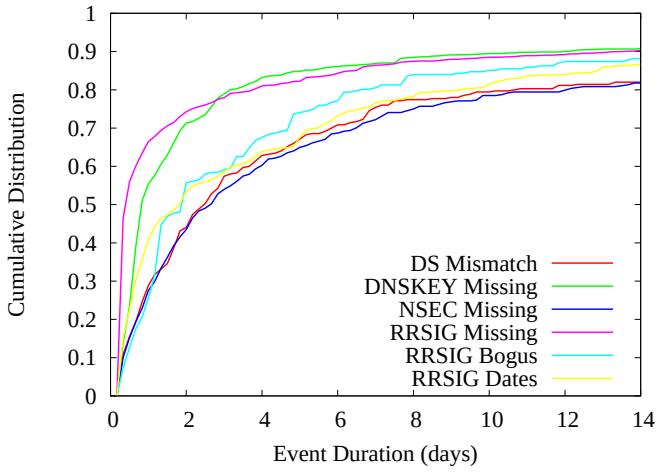

Fig. 6. CDF of the duration of DNSSEC misconfiguration instances of each type.

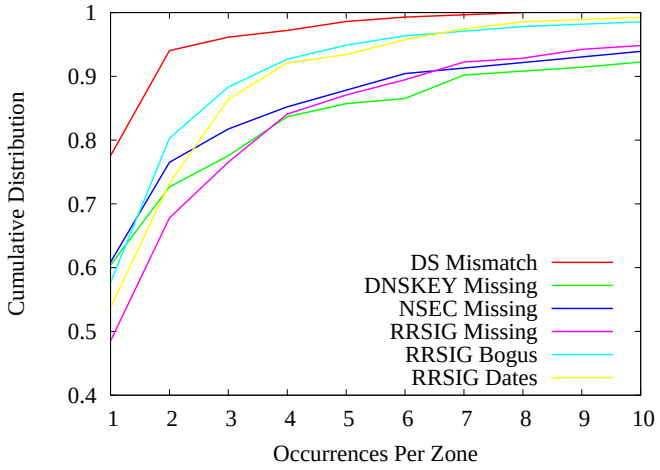

Fig. 7. Number of events per zone, by type.

5 Previous Work

Previous studies have analyzed DNSSEC deployment from various perspectives. SecSpider [3, 18] and IKS Jena [10] discover signed zones and maintain an ongoing status of DNSSEC deployment in terms of pervasiveness and configuration issues. SecSpider monitors from different world-wide locations, assessing consistency of results from different vantage points, and its data was the basis for an assessment of DNSSEC availability, verifiability, and validity [19]. Another DNSSEC availability study has focused on misconfiguration and offered protocol extensions to mitigate their effects [9]. Our work does not seek to examine the breadth of DNSSEC deployment, but focuses on analyzing related patterns of maintenance, misconfiguration, and corrective action in DNSSEC deployment.

6 Conclusion

The DNS is an essential component of the Internet. DNSSEC was designed to protect the integrity of DNS responses, but its deployment has been wrought with challenges. In this paper we have presented an analysis of DNSSEC deployment based on a survey spanning over one year. We observed maintenance behaviors and identified and analyzed configuration errors affecting DNSSEC validation. We analyzed patterns of corrective action and the time taken to correct misconfiguration. We found that misconfiguration affected DNSSEC deployment at even the highest levels and took up to two weeks on average to correct, depending its nature.

We attribute much of the misconfiguration experienced in our survey to the novelty of DNSSEC deployment. DNS administrators are learning the protocol and how to deploy, maintain, and troubleshoot problems. DNSSEC functionality is still fairly new to name server implementations, and they have experienced growing pains. Some implementations have only partial or no support for DNSSEC and yet are serving signed zones, which accounts for some of the instances of partial misconfiguration observed in our survey. Some zones experience significant lapses in zone propagation, resulting in stale data on a subset of servers—a problem which is much less transparent in a DNSSEC environment because of the temporal nature of DNSSEC-related records.

Tools have been developed to help administrators monitor and analyze their DNSSEC deployments, facilitating better understanding and success in deployment. DNSViz [20] provides comprehensive analysis of domain names via a Web interface by analyzing the chain of trust from RRset to anchor, and it checks authoritative servers for consistency. Results are presented to users in a graphical format. The DNSSEC Debugger [23] also traces the chain of trust and provides a detailed textual output of the results. Other online tools examine only the name itself for configuration correctness [1, 4].

If DNSSEC deployment is to be successful, then the challenges posed by its administrative and protocol complexity must be met. Growing experience should be coupled with increased protocol awareness, facilitated by functionality

provided by tools and name sever implementations to achieve a reliable and trustworthy DNS.

References

1. DNSCheck by .SE. http://dnscheck.iis.se/.
2. Root DNSSEC. http://www.root-dnssec.org/.
3. SecSpider. http://secspider.cs.ucla.edu/.
4. Zonecheck. http://www.zonecheck.fr/.
5. R. Arends, R. Austein, M. Larson, D. Massey, and S. Rose. RFC 4033: DNS security introduction and requirements, 2005.
6. R. Arends, R. Austein, M. Larson, D. Massey, and S. Rose. RFC 4034: Resource records for the DNS security extensions, 2005.
7. R. Arends, R. Austein, M. Larson, D. Massey, and S. Rose. RFC 4035: Protocol modifications for the DNS security extensions, 2005.
8. Ramaswamy Chandramouli and Scott Rose. Secure Domain Name System (DNS) Deployment Guide, April 2010.
9. Casey Deccio, Jeff Sedayao, Krishna Kant, and Prasant Mohapatra. Quantifying and improving dnssec availability. In *Proceedings of the 20th International Conference on Computer Communication Networks (ICCCN 2011)*, August 2011.
10. IKS. DNSSEC. https://www.iks-jena.de/leistungen/dnssec.php.
11. Internet Systems Consortium. DNSSEC look-aside validation registry. https://dlv.isc.org/.
12. O. Kolkman and R. Gieben. RFC 4641: DNSSEC operational practices, 2006.
13. B. Laurie, G. Sisson, R. Arends, and D. Blacka. RFC 5155: DNS security (DNSSEC) hashed authenticated denial of existence, 2008.
14. P. Mockapetris. RFC 1034: Domain names - concepts and facilities, 1987.
15. P. Mockapetris. RFC 1035: domain names - implementation and specification, 1987.
16. National Institude of Standards and Technology. Secure naming infrastructure pilot. http://www.dnsops.gov/.
17. Open Directory Project. http://www.dmoz.org/.
18. Eric Osterweil, Dan Massey, and Lixia Zhang. Deploying and monitoring DNS security (DNSSEC). In *25th Annual Computer Security Applications Conference (ACSAC '09)*, December 2009.
19. Eric Osterweil, Michael Ryan, Dan Massey, and Lixia Zhang. Quantifying the operational status of the DNSSEC deployment. In *Proceedings of the 6th ACM/USENIX Internet Measurement Conference (IMC'08)*, October 2008.
20. Sandia National Laboratories. DNSViz. http://dnsviz.net/.
21. SC08: The International Conference for High-performance Computing, Networking, Storage and Analysis. http://sc08.supercomputing.org/.
22. M. StJohns. RFC 5011: Automated updates of DNS security (DNSSEC) trust anchors, 2007.
23. VeriSign Labs. DNSSEC Debugger. http://dnssec-debugger.verisignlabs.com/.
24. S. Weiler. RFC 5074: DNSSEC lookaside validation, 2007.