Security Informatics
a SpringerOpen Journal

RESEARCH Open Access

A measurement study of DNSSEC misconfigurations

Niels L. M. van Adrichem*, Norbert Blenn, Antonio Reyes Lúa, Xin Wang, Muhammad Wasif, Ficky Fatturrahman and Fernando A. Kuipers

Abstract

DNSSEC offers protection against spoofing of DNS data by providing origin authentication, ensuring data integrity and authentication of non-existence by using public-key cryptography. Although the relevance of securing a technology as crucial to the Internet as DNS is obvious, the DNSSEC implementation increases the complexity of the deployed DNS infrastructure, which may result in misconfiguration. In this article, we measure and analyze the misconfigurations for domains in six zones (.bg, .br, .co, .com, .nl and .se). Furthermore, we categorize these misconfigurations and provide an explanation for their possible causes. Finally, we evaluate the effects of misconfigurations on the reachability of a zone's network. Our results show that, although progress has been made in the implementation of DNSSEC, over 4 % of evaluated domains show misconfigurations. The domains with the most frequently appearing misconfiguration are often hosted at a very limited set of hosting providers. Of these misconfigured domains, almost 75 % were unreachable from a DNSSEC-aware resolver. This illustrates that although the authorities of a domain may think their DNS is secured, it is in fact not. Worse still, misconfigured domains are at risk of being unreachable from the clients who care about and implement DNSSEC verification, while the publisher may remain unaware of the error and its consequences.

Keywords: DNS, DNSSEC, Domain Name System, Authentication, Integrity, Misconfiguration, Validation, Signatures, Error, Unreachability

Introduction

The Domain Name System (DNS) [2] is a crucial technology for the functioning of the Internet as it enables communication using domain names that are easier to remember than numerical IP addresses. Among others, DNS maps human-readable hostnames to IP addresses and provides a distributed database from which users can request these mappings. The popularity of this mapping system is explained by the use of Fully Qualified Domain Names (FQDNs) as the primary component of URLs with which all Internet users identify websites.

The importance of DNS lies in the fact that it is not only useful to end users, but that it is also essential to several other core network technologies [3], such as telephone number mapping (ENUM), SIP, email, spam filtering and Microsoft's Active Directory for Windows. However, even though DNS is one of the fundamental building blocks of the Internet, its original design from 1983 focused on scalability and did not include security considerations. Even as early as 1990, the first flaws in the DNS were detected and the need for protection was discussed [4]. The Domain Name System Security Extensions (DNSSEC) were published in 1997 and their refinement in 2005 [5].

DNSSEC, in a broad sense, offers *protection against illegitimately falsifying data stored in the DNS* by providing origin authentication, ensuring integrity and authentication of non-existence. To make sure that the user receives authentic replies, DNSSEC deploys cryptographic keys. With private keys, digital signatures are generated for resources which can be verified by their public counterparts.

*Correspondence: n.l.m.vanadrichem@tudelft.nl
Network Architectures and Services, Delft University of Technology, Mekelweg 4, 2628 CD Delft, The Netherlands

Motivation and problem definition

At the time of writing, there are 115,807,705 .com domains that are configured within name servers, of which 422,037 are signed [6]. For DNSSEC to work as intended, deployment has to span all levels of the DNS architecture. Adoption by all involved actors in the DNS resolution process is therefore essential for success. One big step was given in July 2010 when the DNS root zone was signed [7]. Since then, resolvers are enabled to configure the root zone as a trusted anchor which allows the validation of the complete chain of trust for the first time. Nevertheless, even though 84 % of existing domains could already be using DNSSEC, as more and more Top Level Domains (TLD) are being signed, less than 1 % of authoritative domain name servers have implemented it [8]. Most commonly mentioned causes for this small percentage are:

- the implementation of DNSSEC increases the complexity for the management of the deployed DNS infrastructure,
- a misconfiguration might result in Internet users being unable to reach the protected network [9].

The effects of misconfiguration were very recently made apparent, when a new service from a major television network was unreachable by a significant number of end users due to DNSSEC misconfiguration. Furthermore, the television network resolved the problem by completely disabling DNSSEC authentication for their zone [10]. These types of failures and resulting unreachability of services are not rare incidents, but appear quite often [11]. In fact, there is an IETF Internet-Draft working on clarifying the authoritative domain's responsibility towards a correct DNSSEC configuration [12].

Besides authoritative domains, DNSSEC validation failures have also been reported for complete TLDs [13]. Such problems may occur more frequently when a new group of TLDs (up to 1400) [14] start rolling out within the next few years, since all of them must implement DNSSEC and misconfigurations of their zone files could potentially hide them from the Internet.

Despite the importance of the stated problem, there does not exist sufficient information on the current status of DNSSEC deployed zones. Therefore, in this article, we measure the status of several DNSSEC-enabled operational zones measuring both the level of DNSSEC implementation and the correctness of DNSSEC configuration. We believe that conducting experimental research on DNSSEC is of great value but, in order to get deeper insights, it should also be complemented with an analysis of the most common problems DNSSEC is experiencing in day-to-day production environments. Performing an analysis on real production data from operational zones brings a better understanding on the current status of DNSSEC deployment. Moreover, it also helps to define the biggest challenges that need to be overcome for this technology to succeed.

Outline

The article is organized as follows. "DNS and DNSSEC" summarizes the internal workings of DNS. "Related work" presents related work to misconfigurations, measurements and highlights proposed methods and shortcomings in DNSSEC. In "Measurement tools" the used measurement tools are presented, while "Measurement scenarios" discusses the performed "top-down" and "bottom-up" measurement scenarios.

First, "Results and evaluation: bottom-up approach" presents and analyzes the results of the top-down measurement scenario containing .bg, .br, .co and .se domains. The domains are, among others, chosen as they are browsable through zone walking.[1] Furthermore, the .se zone draws importance for analysis as it was the first DNSSEC signed zone [15] in 2005, 5 years prior to the root domain, hence a high implementation is expected. Similarly, .co is also popular for commercial host names, the zone .br has one of the largest DNS registries in South America, while the .bg domain has signed its zone more recently.

Afterwards, "Results and evaluation: top-down approach" presents and analyzes the results from the bottom-up verification approach on a larger dataset containing .bg, .br, .co, .com, .nl and .se domains. Where the former 4 domains are included for means of comparison, .nl and .com are added for respectively having the highest implementation of DNSSEC-enabled domains [16] and being the largest TLD available. A conclusion is drawn in "Conclusion".

DNS and DNSSEC

In this section, we summarize the functions and internal organization of the Domain Name System (DNS) and its Security Extensions (DNSSEC). In "DNS" we discuss how DNS implements a distributed lookup directory. "DNSSEC" discusses the most relevant extensions and possible errors in configuring DNSSEC.

DNS

The Domain Name System (DNS), first standardized in 1983 [2], primarily offers a distributed database storing typed *values* by *name*. Each record is called a resource

[1] The process of zone walking is explained in "Authentication of non-existing resources"

record (RR), often shortly referred to as “record”, containing the following ordered properties:

- The name of the record.
- The type of the record. Popular records include, the A-, AAAA-, MX-, CNAME- and SPF-records, respectively storing an IPv4 address, IPv6 address, the responsible mail exchange hostname for that domain, an alias referring to another hostname and domain-specific rules regarding spam policies according to the Sender Policy Framework protocol.
- The class code, specifying a name space scope. Although multiple codes exist, usage of values different from IN for Internet are uncommon.
- The time-to-live, specifying in seconds how long intermediate or recursive DNS servers may cache that specific record, often defaulting to a zone’s TTL.
- The length of the RDATA field, used for communication protocol implementation.
- The RDATA field, containing the record’s actual stored data.

Most often, DNS is used to request mappings from computer hostnames to IP addresses. In order to support such a high frequency of requests, DNS employs a tree-wise hierarchy in both name and database structure. A so-called Fully Qualified Domain Name (FQDN) consists of multiple name components specifying the location of its records in a tree of databases. Clients, such as home and business PCs, connect to a local recursive (often caching) DNS server that traverses the tree to receive the requested information. In general, the traversed tree-wise structure of DNS consists of 3 layers: (1) The root-layer, a set of name servers named [a-m].root-servers.net. (2) The Top-Level-Domain (TLD) layer. (3) The authoritative layer.

Every recursive DNS server has a “root hints file”, containing a list of all root-server hostnames and IP addresses. For means of load balancing and geographic distribution of requests, anycast addresses are used to deploy multiple servers per hostname. The root-servers themselves do not contain any mappings for FQDNs, but instead refer to Top-Level-Domain (TLD) servers responsible for the requested TLD by replying with an NS-record containing the name server of the next layer and its IP-address in an A-record. The top-most used types of TLDs are generic TLDs such as the domains .com, .net, .edu and .org, as well as country-code TLDs whose last name suffix refers to country-specific sites such as .nl for the Netherlands, .uk for the United Kingdom, etc. These TLD-servers once again refer to the next layer which is generally authoritative for that domain name and returns the requested mapping. Where further recursion is possible, commonly three steps are sufficient. The relation between the name and the place of its records in the distributed tree is summarized in Fig. 1.

DNSSEC

Although DNS has proven to be very scalable, the architecture shows many possibilities for both un- and intended malicious behavior and attacks. It is fairly easy to tune-in and mangle with DNS requests and replies by executing a so-called man-in-the-middle attack, hence

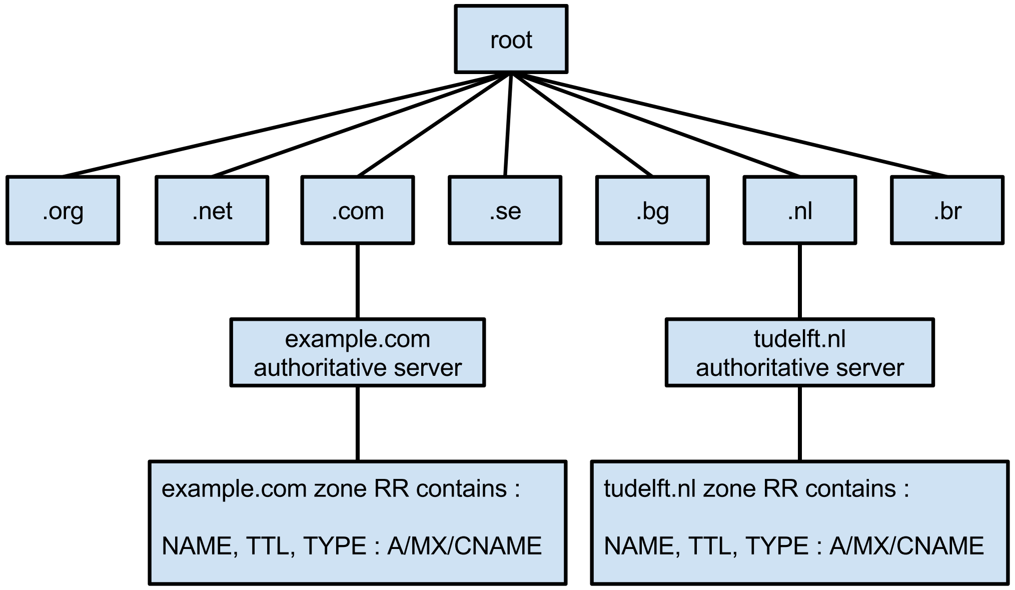
root
.org
.net
.com
.se
.bg
.nl
.br
example.com authoritative server
tudelft.nl authoritative server
example.com zone RR contains :
NAME, TTL, TYPE : A/MX/CNAME
tudelft.nl zone RR contains :
NAME, TTL, TYPE : A/MX/CNAME

Fig. 1 DNS distribution tree. A tree depicting the hierarchical distributed nature of DNS names and their location in the distributed database

secretly redirecting the client to obscure locations, for means of hijacking personal authentication details, or falsely denying existence of resources. This, for example, could occur at open WiFi hotspots, where providers often offer their own, potentially malicious, DNS service. Hence, DNSSEC has been introduced to authenticate the validity of both returned RRs and non-existent records through cryptographic signing of resources.

In order to support the cryptographic signing process, each domain has multiple associated keys containing at least 1 public-private key pair. For each record set (RRset) of distinct name and type, a signature is generated using the private key and stored in an RRSIG-record, which can be verified using the domain's public key stored in a DNSKEY-record, both placed in the zone of the domain.

To confirm the authenticity of the DNSKEY, which is essential to check the authenticity of any record in a zone, its digest, called the Delegation Signer, is stored in a DS-record in its parent zone. Recursively, the DS-record is signed in its local zone, and the process repeats until the root-zone is consulted. Since the root-zone has no ancestors, its DNSKEY-record is confirmed by globally publishing its digest, which is referred to as the Root Trust Anchor [17]. As shown in Fig. 2, a DNS client or resolver recursively requests these records to determine authenticity of a record.

The recursive chain from Trust Anchor, to intermediate DNSKEY-, DS- and RRSIG-records authenticates each RRset of a properly configured DNSSEC domain. Part of managing public-private key pairs is refreshing them regularly to prevent malicious parties from deriving the private key. Hence public-private key pairs are equipped with an expiration date and therefore need to be renewed at regular intervals. Replacing a DNS record, however, is non-trivial due to possibly long periods of caching in clients and intermediate resolvers. Furthermore, changing one's public-key does not only involve updating one's DNSKEY-record, but also implies updating the parent DS-record whose replacement needs to be performed by the TLD, a third party, again keeping cache synchronization in mind. Therefore, key rollover can be a tedious process [18].

In order to ease the process, a zone is equipped with two types of public-private key pairs, Key Signing Keys (KSKs) and Zone Signing Keys (ZSKs), both stored in DNSKEY-records and distinguished by a flag. KSKs concern the keys whose DNSKEY-record is confirmed by the parent's DS-record and are exclusively used to sign other DNSKEY-records in the zone, the ZSKs. ZSKs are hence used to sign all other types of resources in the zone. As ZSKs are signed by a local KSK, those public-private key pairs can rollover independent of the parent-zone. The KSK often is a longer, cryptographically more complex, key pair that changes less often. Besides a decreased need of replacing the key due to its greater complexity, KSKs only sign a limited number of resources (ZSKs) making them less prone to attacks as less in- and output of the key pair is available. The ZSK changes more often and may be cryptographically less complex if sufficiently often replaced with new keys.

Authentication of non-existing resources

When a DNS server is queried for a non-existent record, i.e. there is no record for that requested name, it will respond with a NXDOMAIN message indicating its absence. Where DNSSEC so far authenticates existing resources, it is difficult to authenticate non-existing resources as non-existent records (1) have no corresponding signature, and (2) if for every NXDOMAIN response a signature would be computed online, key pairs would be more vulnerable to attacks. Since an authenticed response for an existing resource may be hijacked and replaced with an NXDOMAIN message, hence falsely denying existence of the resource, it is important to authenticate non-existing resources.

In order to authenticate non-existent resources, DNSSEC introduces NSEC-records [19] containing a linked-list of existing records ordered by name, hence actively denoting non-existing namespaces. For an example domain named example.com with just two subnames mail. and www., the NSEC-records would look as follows:

- The first NSEC-record named example.com refers to mail.example.com indicating that mail.example.com is alphabetically the first subname of example.com, hence actively denying the existence of any subnames that would be ordered prior to it, such as ftp.example.com.
- The second NSEC-record named mail.example.com refers to www.example.com indicating there are alphabetically no subdomain names between them, hence actively denying the existence of subsequently ordered subnames, such as news.example.com.
- The final NSEC-record named www.example.com refers back to the zone name example.com, indicating it is alphabetically the last record.

An NSEC-enabled server will reply with the appropriate NSEC-record to requests for non-existing resources. Each existing NSEC-record, of whom as many exist as names exist for a domain, is ultimately signed by an RRSIG-record to confirm authenticity of the claimed non-existence. A property of the NSEC-records is that they can be iterated to gather a *complete list of valid subnames* for a domain or TLD, a process called

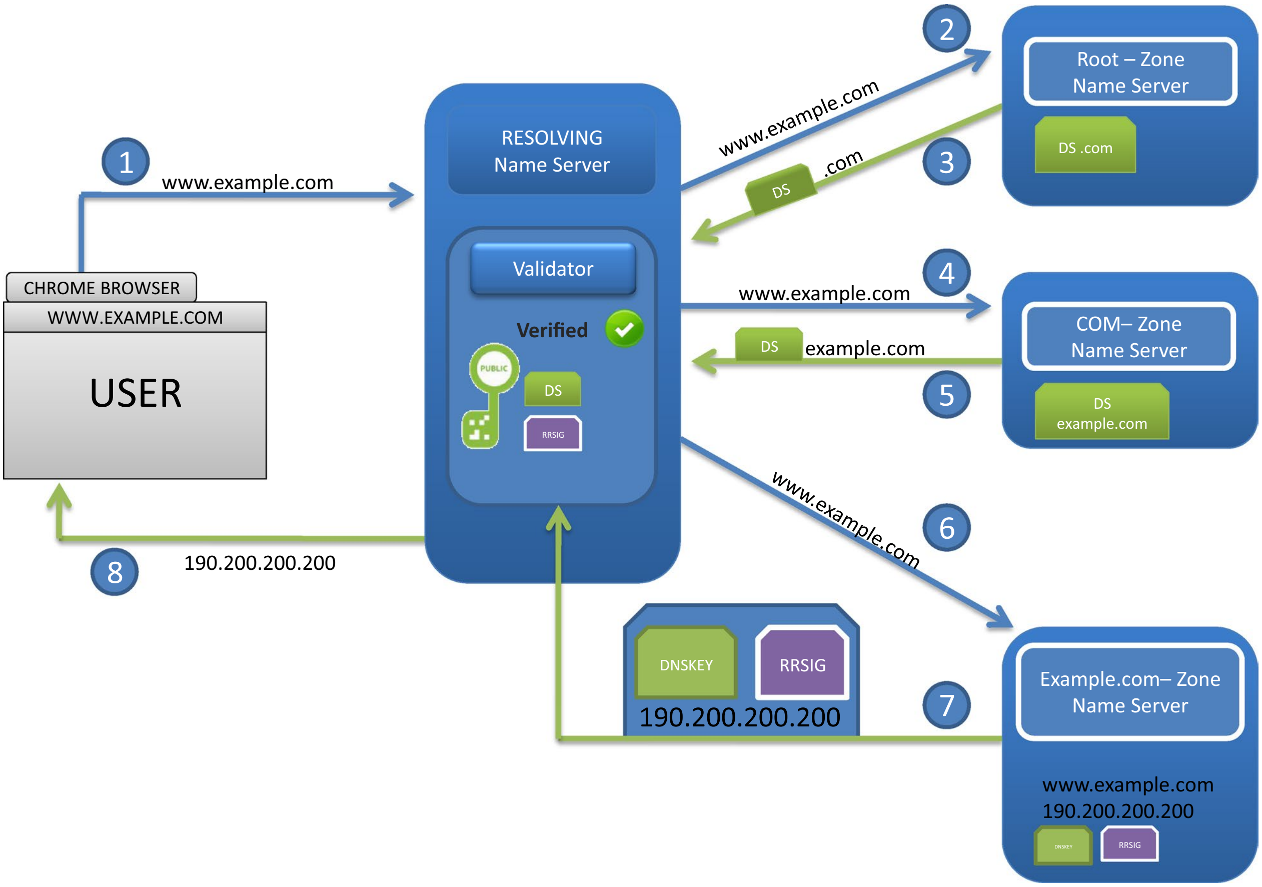

Fig. 2 Domain name resolution in DNSSEC. The domain name query http://www.example.com iterates from the resolver to all the name servers in the DNS tree (steps 2 to 7) until the name is resolved. Between each resolving step all the responses are verified by the validator and, on success, the resolved IP address is forwarded to the user

zone-walking. We have extensively used this method to extract the lists of domain names from our selected TLDs.

However useful to our research, publishing the complete list of available resources does not relate to the occasional request of non-existent resources and may even raise concerns on privacy. Therefore, the improved NSEC3 additions hash the zone-specific name-component (i.e., mail. and www.) prior to ordering, and generate a recursive linked-list of NSEC3-records containing hashed values [20]. In order to authenticate non-existence of a resource, the DNS server will hash the requested zone-specific name component and return the NSEC3-record indicating the non-existent range to which it belongs. Hence, proving non-existence without giving away existing names.

Related work

In this section, we discuss previous work on DNSSEC. We found that most studies focus on the performance of DNSSEC, such as latency, delay, or resource load on the server, rather than on misconfigurations [21]. In [3], the authors analyze the availability of DNSSEC resolution and service, but omit configuration correctness. Lian et al. [22] present a quantitative measurement study towards the capability of resolvers and end-users to perform DNSSEC authentication, which represent the client side of DNSSEC. The authors of [23] present additional security vulnerabilities to DNSSEC itself, while in [24] the security benefits of the NSEC3 hashed authenticated denial of existence[2] are evaluated.

One related study focuses on quantifying and improving DNSSEC availability [9]. The authors first identify what kind of misconfigurations in DNSSEC can affect a DNS query request. They list the potential failures due to DNSSEC misconfiguration, and then they create a metric to quantify those DNSSEC misconfigurations. They classify DNSSEC misconfigurations into three categories:

[2] Summarised in "Authentication of non-existing resources".

1. Zone (missing/expired/invalid RRSIGs covering zone data, or missing DNSKEY RRs required to verify RRSIGs).
2. Delegation (bogus delegations because of lack of appropriate DNSKEYs in the child zone corresponding to DS RRs in the parent zone, or insufficient NSEC RRs to prove an insecure delegation to a resolver).
3. Anchor (stale trust anchors in a resolver, which no longer match appropriate DNSKEYs in the corresponding zone).

They analyze 1456 signed zones out of which 194 show to be misconfigured [25]. Out of these, most of the misconfigurations are related to zone data that correspond to the first class of misconfigurations. However, the authors do not explain why this was the case and what were the main technical causes for such misconfigurations. The class 1 misconfigurations arise due to missing or outdated RRSIGs or DNSKEYs and, as explained in "Introduction", the deployment of those records in the zone file is the responsibility of the zone administrator. Technically, the administrator should always ensure the correctness and validity of RRSIGs and DNSKEY deployment. Hence, the previous work does not give insight in the causes of misconfiguration, nor its effects in authenticity confirmation and reachability by a DNSSEC-aware resolver. Furthermore, the analysis was done in 2010, 5 years ago when DNSSEC was in an earlier stage of deployment compared to now in 2015 when DNSSEC is more widely implemented. In this paper, we analyze over 122,779 signed domains from .bg, .br, .co, .com, .nl and .se domains.

Measurement tools

There exist several different tools to work with DNSSEC. However, most of them are intended to be used by zone administrators in order to verify their own zone file before publishing it and require the user to have the complete zone file in order to perform their tests. Examples of such tools include Verisign's jDNSSEC [26] or NLnetLabs' LDNS [27]. We selected a set of tools that are able to perform tests over a list of several domains without possession of their zone files, that is to say, from the point of view of an external user. Furthermore, we selected these tools based on execution time and ease of automation to ease the process of checking a large amount of domains. We have used the following measurement tools to perform our measurements:

Dnsrecon [28]

A DNS enumeration program, written in Python, that allows to discover relevant information from the content of a zone. It performs several types of enumeration including zone transfer, reverse lookup, a Google lookup and, most importantly, zone walking using NSEC-records as discussed in "Authentication of non-existing resources". After testing the tool, we found its usage straightforward and effective since it provided us with the necessary means to easily and effectively retrieve all the authoritative name servers from a zone.

In the process of retrieving the domain lists for the domain, we enhanced the program with the following 3 functions: (1) To decrease the chance and impact of getting blocked by a DNS server due to excessive usage, we added support for multiple DNS servers. (2) We added the possibility to save the current state of iteration, so in case we were blocked by all DNS servers of a zone we were able to continue iteration from a different source IP-address. (3) Initially, Dnsrecon verified a domain's intent to deploy DNSSEC capability by checking whether the authoritative name server returned a DNSKEY-record for its hostname. Instead, we verified the intent to check whether a domain registered a DS-record at its parent.

DNSSEC-Debugger [29]

A web-based tool developed by Verisign Labs that inspects the chain of trust for a particular DNSSEC-enabled domain. It shows the step-by-step validation of a given domain and indicates any error or warning found in its DNSSEC configuration. We found DNSSEC-Debugger to be fast (analysis consumes approximately 3 s per domain) and ideal for automation as any domain can be inspected by executing a HTTP-request to http://dnssec-debugger.verisignlabs.com/<domain name>, where <domain name> is replaced with the domain name to be analysed. We use DNSSEC-Debugger to perform the top-down validation approach used in our first measurement scenario in "Top-down measurement scenario" and "Results and evaluation: top-down approach".

Google's public DNS service [30]

Found at IP-addresses 8.8.8.8 and 8.8.4.4, Google offers a free and globally accessible DNSSEC-enabled DNS resolution service, which can be used as an alternative to one's in-house or ISP-provided DNS resolution server. In order to evaluate the effects of DNSSEC misconfiguration on the reachability of a domain, we assume a misconfigured DNSSEC domain to be unavailable when it does not pass Google's Public DNS Service.

Dig [31]

Dig (domain information groper), part of the popular DNS server BIND, is a command-line tool that can be used to query DNS servers. It is DNSSEC capable and can be used to verify the DNSSEC chain of trust from a top-down and a bottom-up perspective. However, we found that the current version queries all possible name servers for a TLD or authoritative zone for their A-record, even when glue records are known, when using the top-down approach, resulting in an infeasible amount of lookups. Hence, we only used Dig in a bottom-up approach using a DNSSEC-capable resolver as performed in our second measurement scenario in "Bottom-up measurement scenario".

Measurement scenarios

The DNSSEC chain of trust, consisting of a per-domain zone overlapping chain of public-private key-pairs and signatures as explained in "DNS and DNSSEC", can be verified in two distinct approaches, being:

1. A top-down approach, where first the root zone is verified against the previously known root trust anchor, followed by the TLD zone against its corresponding DS in the root zone, finished by the authoritative zone against its corresponding DS in the TLD zone.
2. A bottom-up approach, where—in reverse order—first the authoritative zone is verified against the TLD zone, which is verified against the root zone, which in turn is verified against the root trust anchor.

In terms of authenticity verification, denoting a record to be either authentic or not, both methods are correct. However, in terms of finding the exact misconfiguration both are incomplete. For example, a top-down approach will assume an authoritative zone to be DNSSEC incapable when it finds no corresponding DS in the TLD zone, while the authoritative zone may serve private-public key pairs and signatures. In such a case, one could argue the domain intends to employ DNSSEC but fails to do so as it did not properly communicate the DS record to be published by the TLD. This specific misconfiguration cannot be found by a top-down approach. The bottom-up approach will find the previously described misconfiguration.

The bottom-up approach, however, assumes an authoritative zone to be DNSSEC incapable when it finds no RRSIG for the record it tries to authenticate. This implies that misconfigurations where the intention to apply DNSSEC by the existence of possible authoritative DNSKEYs or corresponding DSes in the TLD are omitted. To partially solve this problem and get insight in both misconfiguration errors, we have performed two different measurement scenarios on distinct datasets described in "Top-down measurement scenario" and "Bottom-up measurement scenario".

Top-down measurement scenario

The measurement of the different domains in our first measurement scenario consists of 4 different phases, followed by an additional 5th phase in which we evaluate the effects of misconfiguration in everyday use. The first phase consists of gathering a comprehensive list of domain names. To do this, we use Dnsrecon to perform zone-walking of the 4 NSEC-enabled TLDs .bg, .br, .co and .se, hence retrieving extensive lists of domain names from these domains.

The second phase consists of filtering the list of domain names by the intent of them being DNSSEC enabled. We assume a domain name to intend to be DNSSEC enabled when a DS-record for that domain is registered at its TLD-zone. Filtering is performed by iterating the list of domain names and performing DS-record lookups using the internal functions of Dnsrecon.

Having retrieved a list with a sufficient number of DNSSEC-enabled domains, we verify their configuration using the DNSSEC-Debugger online tool from Verisign Labs. To do so, we iterate through the list, performing a HTTP-request to the appropriate URL and parse the response for further analysis. To verify the correctness of the DNSSEC-Debugger, we took a sample from the results and compared them with results from Dig. Normally, it takes approximately 3 s to receive the verification result for one domain name. In order to overcome this time limitation and to speed up the process, we perform up to 10 lookups in parallel by employing multithreading.

In the 4th phase we categorize the misconfiguration in the categories and subcategories denoted in Table 1, enabling analysis by the type of misconfiguration. Besides the expected DNS and DNSSEC related misconfigurations, we found 2 additional errors. We found that a zone's DS could be retracted in the time between retrieving the list of DNSSEC-enabled domains and performing the measurements, meaning the domain has withdrawn from implementing DNSSEC. Furthermore, we found an additional error where the server does not implement the resource record type DNSKEY and, therefore, is DNSSEC incapable.

After performing the initial measurements, we verified the effects of misconfiguration by requesting the A-records associated with misconfigured domains from Google's Public DNS Service which performs DNSSEC authentication verification.

Bottom-up measurement scenario

In our second measurement scenario, we use a publicly available dataset of domain names known as the DNS Census 2013 [32]. The census is published anonymously and contains over 2.6 billion DNS records gathered from over 106 million domain names in 2013. Although the dataset is incomplete and it does not contain official sources, the census can be

Table 1 Top-down measured misconfiguration statistics for the ccTLD .se and respective reachability by Google's public, DNSSEC-aware, DNS resolution service

Error category	Subcategory	Misconfigurations	%	Unreachable	%
DNSKEY	Total	564	64.38	477	84.40
	Not found	261	29.79	259	99.23
	Invalidated by DS	88	10.05	3	3.41
	KSK invalidated by ZSK	215	24.54	214	99.53
	Valid but expired	0	0	N.A.	
RRSIG	Total	1	0.11	1	100.00
	Not found	0	0	N.A.	
	Invalidated by ZSK(s)	0	0	N.A.	
	Valid but expired	1	0.11	1	100.00
	Invalidate RRset	0	0	N.A.	
General DNS failure	Total	295	33.68	163	55.25
	REFUSED	12	1.37	11	91.67
	SERVFAIL	191	21.80	142	74.35
	Time-out	64	7.31	9	14.06
	No SOA	4	0.46	0	0.00
	SOA serial differs	24	2.74	1	4.17
Miscellaneous	Total	16	1.83	7	43.75
	DS retracted	14	1.60	5	35.71
	DNSKEY RR Failure	2	0.23	2	100.00
All categories	Total	876	100	647	73.86

considered significantly large to be representative. From the list of all available domain names we have selected the .bg, .br, .co., .com, .nl and .se TLDs to perform our measurements on. We have chosen the .bg, .br, .co and .se TLDs as those have also been used in our initial measurement scenario, .nl since it has the highest number of DNSSEC implemented domains and .com since it has the largest number of domains in general. The high number of domains in the .com TLD, over 64 million, implies it is difficult to verify all chains of trust within a feasible timespan. To guarantee that the verified subset will be representable, we have randomized the order of the list up front.

Alike to the latter four steps of the previous scenario, we iterate the list of domains and perform DNSSEC validation using Dig and a DNSSEC-capable resolver. Due to the bottom-up approach, authoritative zones are considered DNSSEC capable if they publish RRSIGs for the A-records of their domain name. From the DNSSEC-capable domains, the chain of trust is analyzed and categorized. Finally, the reachability of misconfigured domains is also verified using Google's Public DNS Service.

Results and evaluation: top-down approach

In this section, we discuss and evaluate the results from the experimental measurements described in "Top-down measurement scenario". "DNSSEC implementation" shows the results from the first and second measurement phase, gathering domain names and measuring the integration of DNSSEC in the different zones. "DNSSEC misconfigurations" categorizes the misconfigurations of the zone .se into the type of misconfiguration. Finally, "Effects on availability" analyzes the result of the misconfigurations on the availability of the domain.

DNSSEC implementation

In this subsection, we present the results of the first two phases of our measurements, (1) gathering domain names and (2) measuring the integration of DNSSEC within the list of zones. While gathering the lists of domain names by walking the NSEC-records, we were often blacklisted by the TLD name servers as the excessive amount of performed DNS requests are classified as possible attacks on the service. As shown in Table 2, for most zones we were able to gather and analyze a considerable amount of domain names. The .br zone, however, appeared to have additional counter-measures against zonewalking. Regularly, the .br TLD name servers would reply with an NSEC-record indicating the requested domain was the last domain name of the zone, hence terminating the zone walking process as it appeared to be finished.

Table 2 shows the number of domain names we were able to gather and check for DNSSEC implementation,

Table 2 Top-down measurement on DNSSEC implementation per ccTLD

ccTLD	Retrieved	DNSSEC	%	Misconfigurations	%
.bg	38,806	162	0.42	26	16.04
.br	2481	504	20.31	2	0.00
.co	151,707	23	0.02	6	26.09
.se	89,772	21,748	24.23	876	4.03
Total	282,766	22,437	7.93	910	4.06

Table 3 Historical statistics on DNSSEC implementation per TLD

ccTLD	Statistics from:	Total	DNSSEC	%
.bg	08/2008 [35]	N/A	80	N/A
.br	01/2014 [36]	3,310,972	487,471	14.72
.co	10/2013 [37]	1,560,000	196	0.01
.com	2015 [6]	116,259,506	429,047	0.37
.nl	2013 [16]	5,388,364	1,700,000	31.55
.se	09/2013 [38]	1,292,596	327,684	25.35

while Table 3 shows historical statistics found on per-zone DNSSEC implementation. Although our lists are incomplete, we were able to confirm the relative implementation of DNSSEC for the selected zones. We found that the zones .bg and .co both have a very low implementation of DNSSEC, resulting in a very small set of DNSSEC-enabled domains. For the zone .br, we found a significant amount of DNSSEC-enabled domains. Due to the aforementioned zone-walking counter-measures, however, we were unable to gather a large set of DNSSEC-enabled domains for the .br domain. For the zone .se, we were able to gather an extensive amount of domains and DNSSEC-enabled domains. Hence, we continue to further analyze the configuration mistakes found in the zone .se. Figure 3 shows the relative percentage of DNSSEC implementation per zone.

DNSSEC misconfigurations

As seen in Table 2 and Fig. 4, the ccTLD .se has a significant amount of misconfigurations. Table 1 shows the misconfigurations related to the categories and subcategories listed in "Measurement scenarios". As also prospected in Fig. 5, approximately two-thirds of the misconfigurations are related to configuration of the DNSKEY records. Slightly less than one-third of the misconfigurations are caused by missing DNSKEY records, indicating there once was an intention or maybe even a running configuration to deploy DNSSEC. However, the DNSSEC configuration has never been properly configured or was removed from the authoritative name server. Slightly more than a third of the misconfigurations indicate a Key Signing Key that is not properly signed by its Zone Signing Key as described in "DNSSEC", hence breaking the chain of trust. The situation where a ZSK invalidates the zone's KSK indicates a problem with the key-rollover of the KSK, most probably its signature has not been recomputed after it was renewed. Once DNSKEY configuration is properly done, we find little evidence in the internal configuration of the authoritative name server, from the category of possible RRSIG misconfigurations we only found one occurrence of an expired signature.

Stunningly, a third of the misconfigurations seem to revolve around general DNS misconfigurations or errors that could also occur in non-DNSSEC environments. Especially the number of reported server failures and time-outs are surprisingly high. We were unable to confirm whether these errors are strictly related to non-DNSSEC configuration, and thus unrelated to DNSSEC,

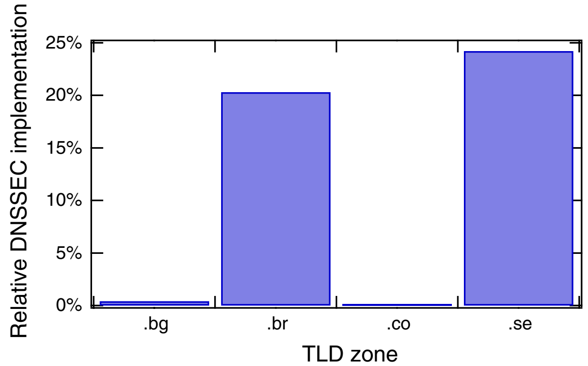

Fig. 3 Top-down per-TLD DNSSEC implementation. Relative implementation of DNSSEC per country-code TLD

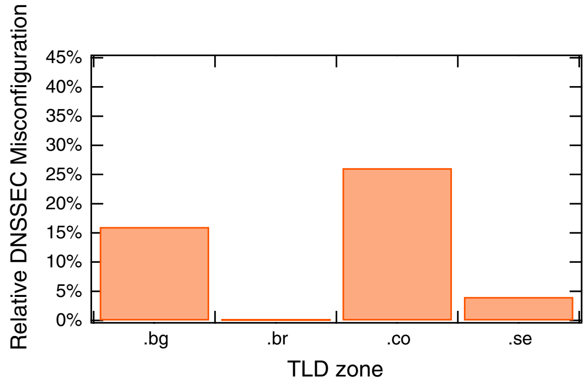

Fig. 4 Top-down DNSSEC misconfigurations. Relative misconfigurations found through top-down verification of DNSSEC domains

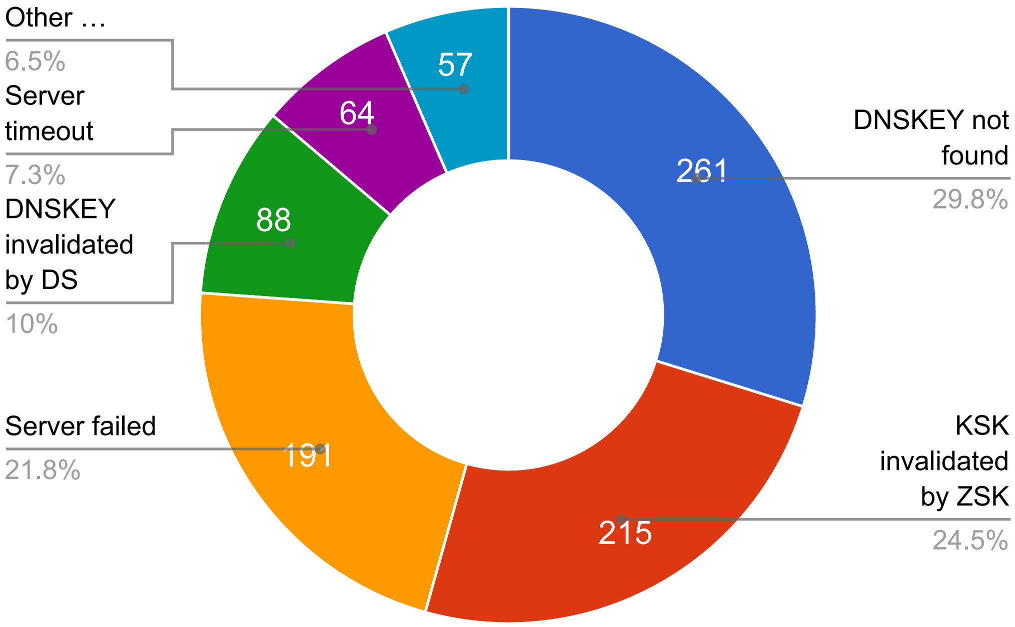

Fig. 5 Top-down DNSSEC misconfiguration distribution. Distribution of most significant DNSSEC misconfigurations

or are caused by a server malfunction due to incompatibility with the DNSSEC-extended query. We did however find two occurrences with a more specific error, where the server indicated incompatibility with the DNSKEY resource record type, showing that DNSSEC-incompatibility with DNS servers once intended to perform DNSSEC is a problem. Finally, we found 14 occurrences in which the authoritative administration retracted its intention to implement DNSSEC before we were able to scan its zone for misconfiguration. In Fig. 5, we show the distribution among the most significant configuration errors.

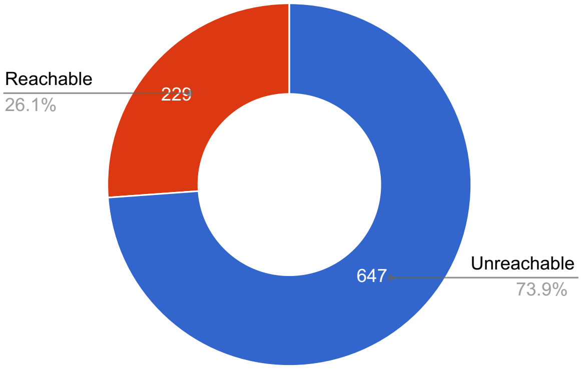

Fig. 6 Domain availability. Availability of misconfigured DNSSEC-enabled domains

Effects on availability

After performing the measurements, we proceeded to verify the reachability of the misconfigured domains using a DNSSEC validating resolver. For that purpose, we used Google's Public DNS Service, which has implemented DNSSEC validation by default since May 2013 [30].

The result of this experiment shows that 73.86 % of the misconfigured domains in the ccTLD .se were completely unreachable from a DNSSEC-aware resolver. As also shown in Fig. 6, the remaining 26.14 % of domains still had some misconfiguration, but those were not as severe to provoke the domain to become unavailable. To learn the impact of a misconfiguration, we correlated the (un-)reachability of each domain to its misconfiguration category from Table 1.

Summarized in Fig. 7, after combining the categories with less than 50 misconfigurations, the impact on reachability of the most common misconfiguration types becomes clear. Concerning the DNSSEC-specific misconfigurations, the impact of a missing DNSKEY record or a ZSK being invalidated by its KSK is large, nearing a 100 % of unreachability. A DNSKEY invalidated by the parent DS-record indicating a potential security breach of the complete domain, however, only fails integrity checks at 3.41 % of the sampled domains, even though this error may be considered as serious as the previous DNSKEY-related errors. A general DNS server failure when the DNSKEY is requested leads to an unreachability level of 74.35%, similar to the overall average. Finally,

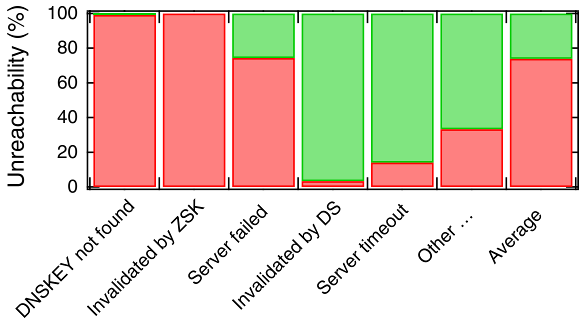

Fig. 7 Top-down per-category misconfigurations. Relative misconfigurations per category in the ccTLD .se found through top-down evaluation

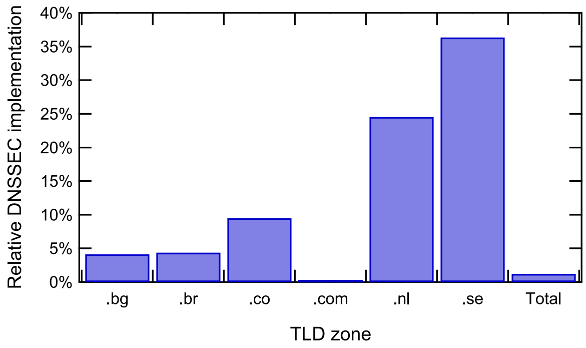

Fig. 8 Bottom-up per-TLD DNSSEC implementation. Relative implementation of DNSSEC found through bottom-up evaluation per TLD

we notice server timeouts are handled correctly in most cases by the caching functionality of the resolver.

Results and evaluation: bottom-up approach

In this section, we discuss and evaluate the results from the experimental measurements described in "Bottom-up measurement scenario". "DNSSEC implementation" shows the results from the first measurement phase, measuring the integration and initial validation of DNSSEC in the different domains. "DNSSEC implementation" further discusses the different categories of found misconfigurations and their implications. In "Analysis of domains without a DS", we perform further experiments on the root cause of the most common misconfiguration mistake. Finally, "Signature validity" discusses additional results we collected on the validity of signatures.

DNSSEC implementation

We were able to traverse all available domains from the .bg, .br, .co, .nl and .se TLDs found in the DNS Census 2013 [32]. Due to its large size, we were only able to retrieve 65.74 % of available .com domains, resulting in over 42 million retrieved domains. Due to upfront randomization of the traversed list, we claim to have retrieved a representable dataset. Table 4 and Fig. 8 show the level of DNSSEC implementation and ratio of found misconfigurations.

Due to the difference in measuring the intent to employ DNSSEC compared to the previous measurement strategy, we find different numbers for the implementation of DNSSEC. Particularly, .bg, .co and .se show much higher numbers of domains implementing DNSSEC due to the inverse direction of detection. Where previously a DS record in the TLD denoted a zone DNSSEC capable, now an existing RRSIG for the domain does. The TLD .br, on the other hand, shows a lower number of implemented number of DNSSEC-capable domains, which may be explained by the small dataset of verified domains in the previous measurement scenario.

Finally, .nl and .com are added to the list of traversed domains. Where .nl shows a high implementation of DNSSEC, the implementation of DNSSEC in .com, the largest TLD, remains excruciatingly low. Furthermore, .com's level of misconfiguration is very high due to a specific misconfiguration explained in the following two subsections.

DNSSEC misconfigurations

Where the bottom-up approach not only results in more domains to be considered DNSSEC capable, it may also change the detected misconfiguration as a domain can be misconfigured at multiple locations in the chain. From the domains that were verified to implement DNSSEC, Table 5 shows the per-TLD validation results and

Table 4 Bottom-up measurement on DNSSEC implementation per TLD

ccTLD	In dataset	Retrieved	DNSSEC	%	Misconfigurations	%
.bg	13,288	13,288	549	4.13	41	7.47
.br	572,506	572,506	25,037	4.37	649	2.59
.co	72,305	72,305	6871	9.50	260	3.78
.com	64,337,635	42,239,548	122,779	0.29	55297	45.04
.nl	1,062,209	1,062,140	260,752	24.55	26278	10.08
.se	352,235	3,52,235	128,008	36.34	53321	41.65
Total	66,410,178	44,312,022	543,996	1.23	135,846	24.97

Table 5 Bottom-up measured misconfiguration statistics per TLD

TLD	Validation successful	No DS found	RRSIG expired	Timeout	Measurement error
.bg	481	28	1	12	27
.br	23,385	403	61	185	1003
.co	6507	89	5	166	104
.com	66,580	38,213	560	16,524	902
.nl	232,334	4789	4124	17,365	2140
.se	63,987	49,094	10	4217	10,700
Total	393,274	92,616	4761	38,469	14,876

the relative configuration distribution is shown in Fig. 9. Apart from the previously witnessed relative increase in DNSSEC implementations, we also witness an increase in misconfigurations within found DNSSEC domains.

The increase in misconfiguration is explained by the fact that where a DNSSEC-capable domain without a DS record would be considered DNSSEC incapable in the top-down scenario, in the bottom-up scenario it is considered a distinct misconfiguration categorized as "No DS Found". In fact, the results show that the main misconfiguration error concerns domains that have no valid DS published in their respective TLD. Since the bottom-up approach first verifies local RRSIG and DNSKEYs before requesting a DS, it implies that further configuration of the zone is correct and the chain of trust is merely broken by this one absent record. Hence, in total over 68 % of the misconfigurations is due to this relatively small error of not communicating one's DS to the TLD.

Furthermore, we find that where the top-down approach showed many different categories of misconfigurations, in the bottom-up approach further errors are reduced to an expired RRSIG or time-out. The high occurrence of expired RRSIGs and absence of errors in DNSKEY validation (and exactly vice versa in the top-down approach) implies that domains that have DNSKEY validation problems (either due to DS- or KSK/ZSK-invalidation) also let their signatures expire. Hence, there was once a valid chain of trust which becomes outdated and invalid due to administrative neglection.

Finally, we had few occurrences in which Dig would crash with unclear errors. For means of completeness, we have added these occurrences as measurement error to our dataset.

Analysis of domains without a DS

As explained in the previous two subsections, many more misconfigurations have been found due to the detection of DNSSEC-capable domains implementing signatures and private-public key pairs that omit communicating their delegate signer to their TLD. Whereas implementing DNSSEC locally may be a matter of configuration and key computation, uploading the DS is a process that is TLD specific and may require periodic renewal. Hence, we find many domains that show this specific error.

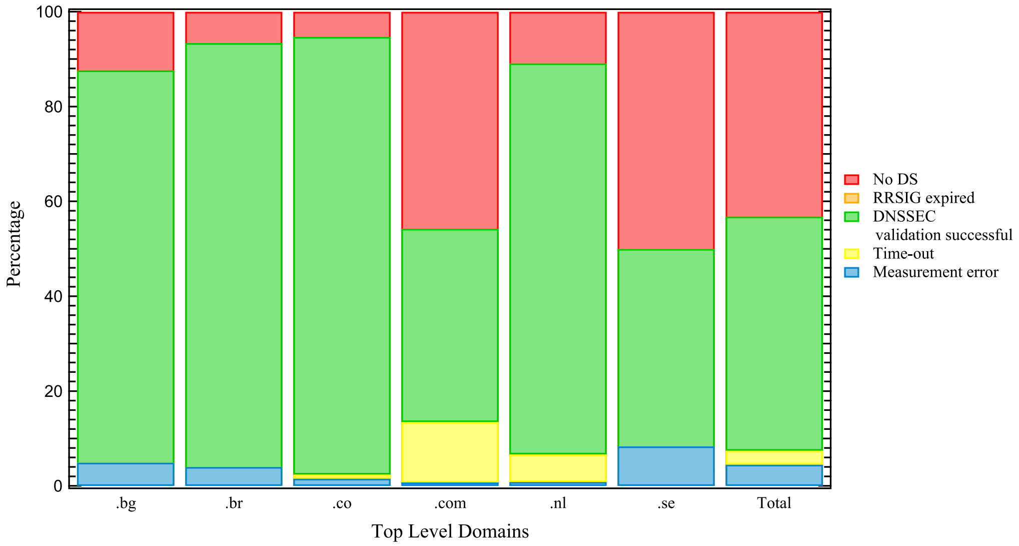

Fig. 9 Bottom-up DNSSEC configuration distribution. Distribution of most significant DNSSEC misconfigurations through bottom-up evaluation

Most owners of a domain do not run their own DNS and web servers, but instead use a web hosting service to arrange those technicalities for them. Therefore, it is interesting to find whether the source of these many misconfigurations share a common cause. Hence, for each domain we have performed a *whois* lookup for the IP address stored in its A-record. A whois lookup is an information request about the owner of an IP address or domain name. For this, we have used the whois host server of Team Cymru [33], which contains mappings from IP address to Autonomous System (AS). We use the AS to identify the web hosting party that serves that particular domain.

Given that we know the technically responsible party for each domain, we have counted the number of DNSSEC-capable domains without DS in its TLD per party. Figure 10 shows the over-representation in percent points from the average, ordered by the level of over-representation. From this figure we derive that the top 5 ASes account for 60.6 % of all domains not having a DS in its TLD to verify it, hence breaking the chain of trust. Hence, *most misconfigurations are caused by a very limited set of web hosting parties*. This conclusion is further backed up by the fact that the 35,153,800 domains containing A-records refer to only 3,370,503 unique IP addresses, indicating many providers deploy virtual hosting by having a single webserver host many websites. Once such a service is affected by a configuration error, all of its domains are.

Signature validity

Additionally, we were able to measure the validity duration of the RRSIGs of the found traversed names by subtracting the expiration field by its inception field. Fig. 11 shows the probability distribution function of the measured validity periods. A 3-week validity period is most common, followed by a (short) 2 week period. Although a few shorter intervals occur, round numbers such as half a year (183 days), 1 month (30 days) and 40 days are most popular. Interestingly, we found 1 signature that wouldn't expire until 69 years after its inception. This value conforms to the maximum duration implied by the compulsory use of serial number arithmetic [34], which is necessary to wrap-around the maximal available date and time of 2^{32} s (around 136 years) after 1 January 1970 00:00:00 UTC.

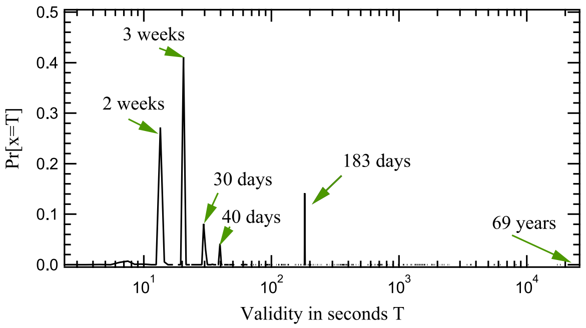

Fig. 11 Signature validity. The probability distribution function of signature validity

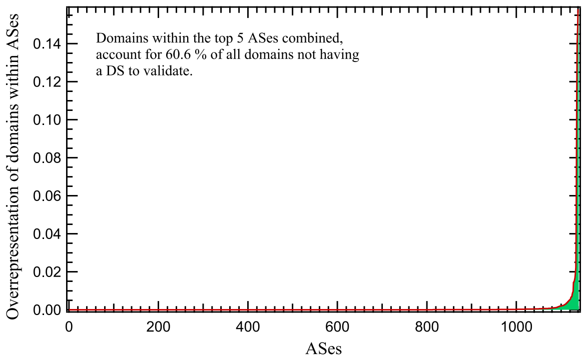

Fig. 10 Overrepresented ASes showing missing DSes. Ordered distribution of ASes overrepresented in the misconfiguration category of missing DSes

Conclusion

Having analyzed the DNSSEC misconfigurations of four zones (.bg, .br., co. and .se), our measurements show that implementing DNSSEC is not trivial and that misconfigurations exist in large numbers. From the 282,766 gathered domain names, only 7.93 % show the intent to implement DNSSEC. Furthermore, over 4 % of DNSSEC-enabled domains show a form of misconfiguration, emphasizing the configuration complexity. Where one might expect expiration of keys to be a significant means of misconfiguration, categorization of the errors found in the .se domains shows its impact to be neglectable. Instead, most DNSSEC-related misconfigurations are caused by an inconsistency concerning the DNSKEY, the main public key of a domain. In more than 99 % of the cases of a missing DNSKEY or an error in the two-stage ZSK and KSK DNSKEY signing process, the error led to an unreachable domain and thus unreachable website or other network service. User availability shows to vary per type of misconfiguration. On average, 73.86 % of the misconfigured domains appeared unreachable from a DNSSEC-aware resolver. Hence, organizations implementing DNSSEC need to frequently verify the correct configuration of DNSSEC parameters and perhaps implement mechanisms to guarantee continuous correctness of configuration and authentic availability of their resources.

Measurements on a second dataset, including domains from .bg, .br, .co, .com, .nl and .se, show that many more domains have the intent of implementing DNSSEC (over 20 % excluding .com) but fail to communicate an essential part to their respective TLD. We show that a limited number of faulty configured web hosting parties cause most of the misconfigurations. Furthermore, we observe that where signature expirations were initially practically absent, now we exclusively find expired signatures. This implies that key invalidation almost always coincides with signature expiration, both results of a neglected zone. Furthermore, mapping the misconfigured domains to their respective web hosting provider shows that only 5 providers are responsible for 60.6 % of the found misconfigurations. Finally, we show that a 3 and 2 week validity period between inception and expiration are most popular, followed by half-year and 1 month renewal periods.

Acknowledgements

The authors thank Christian Doerr for his valuable feedback and participation in insightful discussions. Furthermore, we thank Verisign Labs for providing us with detailed information on the verification tests performed by the Verisign DNSSEC Debugger. This research has been partly supported by the EU FP7 Network of Excellence in Internet Science EINS (project no. 288021). This article is an extended version of van Adrichem et al. [1].

Competing interests

The authors declare that they have no competing interests.

Received: 13 March 2015 Accepted: 7 October 2015
Published online: 19 October 2015

References

1. van Adrichem NLM, Lúa AR, Wang X, Wasif M, Fatturrahman F, Kuipers FA (2014) DNSSEC Misconfigurations: how incorrectly configured security leads to unreachability. In: Intelligence and Security Informatics Conference (JISIC), 2014 IEEE Joint, pp 9–16
2. Mockapetris PV (1987) Domain Names—Implementation and Specification. IETF. RFC 1035 (INTERNET STANDARD). http://www.ietf.org/rfc/rfc1035.txt
3. Migault D, Girard C, Laurent M (2010) A Performance view on DNSSEC migration. In: 2010 International Conference on Network and Service Management (CNSM), IEEE, pp 469–474
4. Bellovin SM (1995) Using the Domain Name System for System Break-ins. In: Proceedings of 5th USENIX UNIX Security Symposium, USENIX Association, Berkeley
5. Arends R, Austein R, Larson M, Massey D, Rose S (2005) DNS Security - Introduction and Requirements. IETF. RFC 4033 (Proposed Standard). http://www.ietf.org/rfc/rfc4033.txt
6. Cambus F StatDNS—DNS and Domain Name statistics. http://www.statdns.com. Accessed 02 Mar 2015
7. ICANN and VeriSign Inc.: Root DNSSEC. http://www.root-dnssec.org/. Accessed 02 Mar 2015
8. ICANN: DNSSEC Securing the Internet: benefits to Companies and Consumers. http://www.icann.org/en/news/in-focus/dnssec/dnssec-card-03dec12-en.pdf. Accessed 02 Mar 2015
9. Deccio C, Sedayao J, Kant K, Mohapatra P (2011) Quantifying and Improving DNSSEC Availability. In: Proceedings of 20th International Conference On Computer Communications and Networks (ICCCN), 2011, pp 1–7
10. York D. HBO NOW DNSSEC Misconfiguration Makes Site Unavailable From Comcast Networks (Fixed Now) Deploy360 Programme. http://www.internetsociety.org/deploy360/blog/2015/03/hbo-now-dnssec-misconfiguration-makes-site-unavailable-from-comcast-networks-fixed-now/. Accessed 12 May 2015
11. Comcast: DNSSEC News. http://dns.comcast.net/index.php/categories/listings/dnssec-news. Accessed 22 Jul 2015
12. Livingood J (2015) Responsibility for Authoritative DNSSEC Mistakes. Working Draft. http://www.ietf.org/internet-drafts/draft-livingood-dnsop-auth-dnssec-mistakes-02.txt
13. Comcast: gov Failing DNSSEC Validation. http://dns.comcast.net/index.php/entry/gov-failing-dnssec-validation-1. Accessed 22 Jul 2015
14. ICANN: Applicant Guidebook | ICANN New gTLDs. http://newgtlds.icann.org/en/applicants/agb. Accessed 02 Mar 2015
15. Internet Infrastructure Foundation: DNSSEC—The Path to a Secure Domain. https://www.iis.se/english/domains/tech/dnssec/. Accessed 02 Mar 2015
16. SIDN: SIDN Annual Report 2013. https://www.sidn.nl/annualreport/dot-nl. Accessed 04 Mar 2015
17. IANA: Root Zone DNSSEC Trust Anchors. http://data.iana.org/root-anchors/. Accessed 02 Mar 2015
18. Schaeffer Y, Overeinder B, Mekking M (2012) Flexible and Robust Key Rollover in DNSSEC. In: Proceedings of the Workshop on Securing and Trusting Internet Names (SATIN 2012). NPL
19. Schlyter J (2004) DNS Security (DNSSEC) NextSECure (NSEC) RDATA Format. IETF. RFC 3845 (Proposed Standard). http://www.ietf.org/rfc/rfc3845.txt
20. Laurie B, Sisson G, Arends R, Blacka D (2008) DNS Security (DNSSEC) Hashed Authenticated Denial of Existence. IETF. RFC 5155 (Proposed Standard). http://www.ietf.org/rfc/rfc5155.txt
21. Huston G, Michaelson G. Measuring DNSSEC Performance. http://impossible.rand.apnic.net/ispcol/2013-05/dnssec-performance.pdf. Accessed 22 Jul 2015
22. Lian W, Rescorla E, Shacham H, Savage S (2013) Measuring the Practical Impact of DNSSEC Deployment. In: USENIX Security, pp 573–588
23. Ariyapperuma S, Mitchell CJ (2007) Security vulnerabilities in DNS and DNSSEC. In: The Second International Conference On Availability, Reliability and Security, 2007. ARES 2007. pp 335–342
24. Bau J, Mitchell JC (2010) A Security Evaluation of DNSSEC with NSEC3. IACR Cryptol ePrint Archiv 2010:115
25. Deccio C (2010) Quantifying the Impact of DNSSEC Misconfiguration. In: DNS-OARC Workshop (2). https://www.dns-oarc.net/files/workshop-201010/Casey-2010-10-14-dns-oarc-orig.pdf
26. Verisign Labs and Internet Innovations: DNSSEC Analyzer—jdnssec-tools. http://www.verisignlabs.com/jdnssec-tools/. Accessed 02 Mar 2015
27. NLnet Labs: ldns. http://www.nlnetlabs.nl/projects/ldns/. Accessed 02 Mar 2015
28. Perez C Dnsrecon. https://github.com/darkoperator/dnsrecon. Accessed Jan 2014
29. Verisign Labs and Internet Innovations: DNSSEC Analyzer. http://Dnssec-debugger.verisignlabs.com. Accessed 02 Mar 2015
30. Google: Public DNS—Google Developers. https://developers.google.com/speed/public-dns/. Accessed 02 Mar 2015
31. BIND: dig (domain information groper). ftp://ftp.isc.org/isc/bind9/cur/9.10/doc/arm/man.dig.html. Accessed 04 Mar 2015
32. Anonymous: DNS Census 2013. https://dnscensus2013.neocities.org/. Accessed 10 Mar 2015
33. Team Cymru. IP to ASN Mapping. https://www.team-cymru.org/IP-ASN-mapping.html. Accessed 10 Mar 2015
34. Elz R, Bush R (1996) Serial Number Arithmetic. IETF. RFC 1982 (Proposed Standard) (1996). http://www.ietf.org/rfc/rfc1982.txt
35. Kalchev D DNSSEC Implementation in .BG. http://www.cctld.ru/files/pdf/Presentations_09-09/17-45_dnssec%20Bulgaria.pdf. Accessed Jan 2014
36. Home-Dominios-Estatisticas. http://registro.br/estatisticas.html. Accessed Jan 2014
37. Romero G (2013) DNSSEC Deployment in .CO. In: ICANN48 DNSSEC Workshop
38. Internet Infrastructure Foundation: Growth .se | .SE. https://www.iis.se/english/domains/domain-statistics/growth/. Accessed 01 2014