Best Practices for Resilience of Authoritative DNS Servers

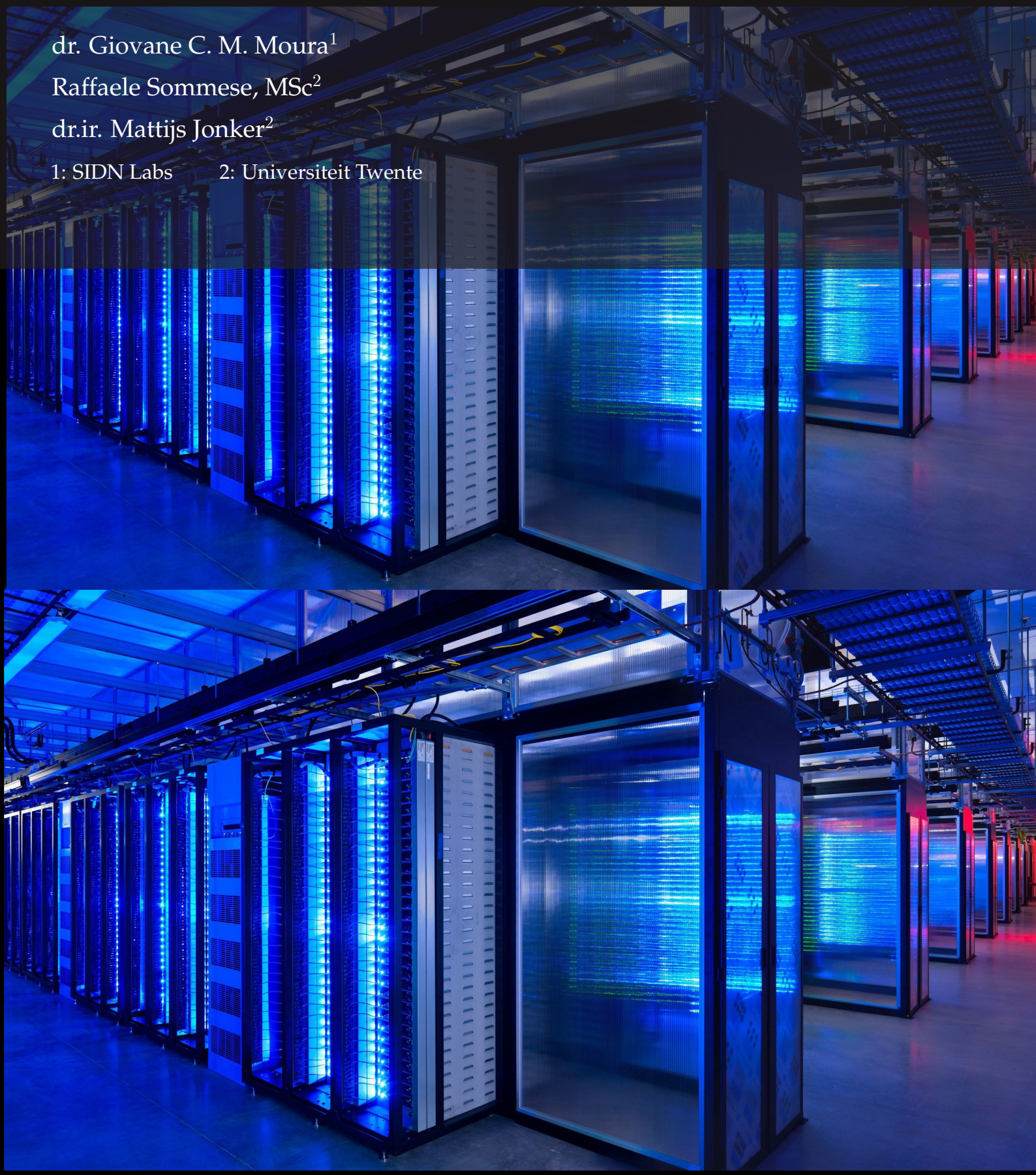
DINO Project

dr. Giovane C. M. Moura[1]

Raffaele Sommese, MSc[2]

dr.ir. Mattijs Jonker[2]

1: SIDN Labs 2: Universiteit Twente



Best Practices for Resilience of Authoritative DNS Servers

DINO Project

by

dr. Giovane C. M. Moura[1]
Raffaele Sommese, MSc[2]
dr.ir. Mattijs Jonker[2]

1: SIDN Labs 2: Universiteit Twente

Thursday 17th March, 2022

Abstract

This document fulfills Task 1 (T1) from the plan van aanpak (PvA). We identify and describe best practices that, if implemented by DNS operators, bring about resilience for authoritative nameservers. These best practices will be used as a starting point in a later task, in which we investigate the extent to which these best practices are currently adhered to by operators of DNS infrastructure associated with governmental services and therefore vitally important to the Netherlands society.

1

Introduction

The Internet Domain Name System (DNS) [1] is one of the core services on the Internet. It maps servers, resources, and services to IP addresses. Every web page visit requires a series of DNS queries, and large-scale DNS failures can have global, cascading effects. DNS-related incidents can make the front pages of prominent news outlets, as in the case of denial-of-service (DDoS) attack agains Dyn DNS in 2016. In this particular incident, the Mirai botnet [2] was used to overload the Authoritative servers of Dyn, compromising the reachability of various proeminent websites, such as Netlifx, Spotify, Reddit and the New York times [3].

2

Background

We provide brief background information here on the DNS and its components to help put some of the best practices that we later identify in context.

2.1. Types of DNS servers

The DNS is a distributed and hierarchical system. It can be seen as a distributed database, in which the management and operation of parts of that database can be delegated for technical and administrative scalability. In general terms, the DNS involves two types of servers, as we show in Figure 2.1. Authoritative DNS servers, which are the focus of this work and in green in the figure, are servers that are – as the name suggests – authoritative for a part of the global DNS hierarchy. These servers know the contents they are responsible for from memory [4]. As an example, `ns1.dns.nl` is one of the authoritative servers for the `.nl` zone. This server knows where to find other authoritative servers that are responsible for smaller parts of the `.nl` zone, i.e., domain names further *down* the DNS hierarchy.

DNS resolvers, in turn (salmon color in Figure 2.1), are servers that, on behalf and users and applications, perform the task of looking up information in the DNS. As an example task, consider resolving a domain name to an IP address. Because of the hierarchical approach, such resolvers *recursively* query the DNS. That is, they potentially reach out to authoritatives in various layers of the DNS hierarchy.

As a concrete example, if a user (shown as stub in the figure) wants to visit `wikipedia.org` in their browser, she first needs to use one of her DNS resolvers to retrieve the IP address of this domain name. The resolver, in turn, will attempt to resolve the domain and ultimately obtain a response from

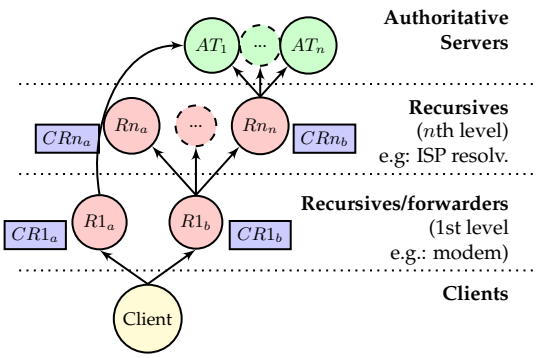


Figure 2.1: Relationship between clients (yellow), recursive resolvers (red) with their caches (blue), and authoritative servers (green).

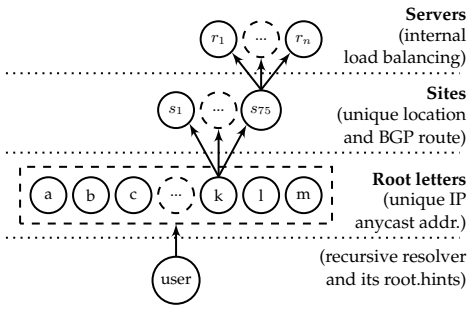


Figure 2.2: Root DNS structure, terminology, and mechanisms in use at each level.

the authoritative DNS server for `wikipedia.org` (`ns[1-3].wikimedia.org`), which will then send the requested IP address back to the user.

2.2. Authoritative DNS servers setup and redundancy

Any DNS zone (such as `example.org`) must be configured with authoritative DNS servers, which are the servers that can respond DNS queries from resolvers. These authoritative servers are defined in so-called NS records [1] in the DNS.

Replication of a DNS service is important to support high reliability and capacity and to reduce latency. The DNS has two complementary mechanisms to replicate service. First, the protocol itself supports *nameserver replication* of DNS service for a zone, by supporting multiple NS records for a given zone. Figure 2.2 shows the setup of the Root DNS zone (`.`), which has 13 authoritative DNS servers (`[a-m].root-servers.net.`). Each of these NS records have their own IPv4 and IPv6 addresses, defined as A and AAAA resource records.

Second, each of these authoritative servers can run in multiple physical locations while using *IP anycast* [5, 6]. This is different from the aforementioned replication through multiple NS records, because in the anycast case the same IP address is shared between physical locations, while the Internet routing (BGP) is leveraged to direct clients to the nearest anycast site. Note that a combination between both mechanisms – multiple nameservers and multiple physical locations for each nameserver – is also possible.

Nameserver replication is recommended for all zones, and IP anycast is used by most large zones such as the DNS Root and most top-level domains [7, 8]. IP anycast is also widely used by *public resolvers*, which are DNS resolvers that are open for use by anyone on the Internet. As examples, consider Google Public DNS [9], OpenDNS [10], Quad9 [11], and `1.1.1.1` [12]. In the root zone (Figure 2.2), we show that `K-ROOT`, one of the root authoritative servers ran by RIPE NCC, has 75 anycast sites (S_n). BGP [13] then *maps* the IPv4 and IPv6 clients to individual sites and, in this way, a DDoS attacks can have limited effect by overwhelming *some* of the sites while leaving others active [8].

Finally, the last level of replication is per anycast site, in which each *site* can have multiple servers behind a load balancers (r_n) in Figure 2.2. (Unicast servers can also have load balancers, but they have a single site).

3

Best Practices

In the section we present best-practices on how to configure DNS authoritative servers. It summarizes the conclusions from these research efforts and offers specific, tangible advice to operators when configuring authoritative DNS servers.

We divide the best practices into three categories: *critical* and *recommended*, and *immeasurable*. Critical (§3.1) refers to practices that are a *must* to overcome *single points-of-failure* (SPoF) – analogous to "don't put all your eggs in the same basket". Single points-of-failure cause total unreachability of domain names when they fail.

The second category of best practices are *recommended* (§3.2), which means that they *help* to improve the resilience of DNS, but not following them does not lead to single points-of-failure.

The last category are best practices that we consider out-of-scope of this study (§3.2). There are practices that cannot be measured using traditional Internet Measurements (layer 3 and above), such as physical and link layer practices. We however list them given their importance, although we cannot access them in this study.

We note that following these practices may imply more financial costs. For example, hosting authoritative DNS servers on multiple Autonomous Systems may cost more than hosting on a single AS. We, however, do not take *costs* into consideration, but will mention where they could be significantly higher.

These best practices concern *availablity* of a DNS zone and not its integrity. In this sense, we focus on metrics and properties that could improve the dependability and availablity of authoritative servers. We do not, however, focus on best practices not related to availablity, such as use of DNSSEC [14] that guarantees DNS messages authenticity and integrity and best pratices to reduce latency between clients and authoritative servers (performance).

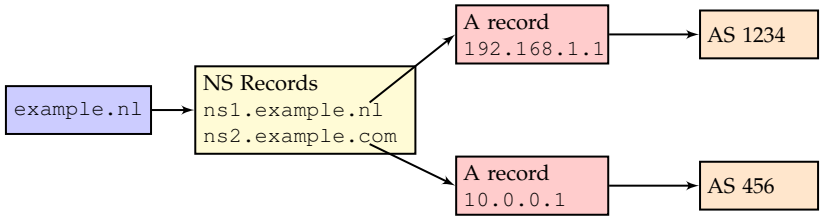


Figure 3.1: Example to illustate best practices for an example domain name (`example.nl`)

Metric	Description/Reference	Value
nNSes	Number of NS records for a zone/[15]	>=2
nIP(NSv4)	Number of Unique IP addresses for NSes (IPv4) [15]	>=2
nIP(NSv6)	Number of Unique IP addresses for NSes (IPv6) [15]	>=2
ResponsiveNSesV4	All authoritative servers are responsive for the domain/[16]	`True`
ResponsiveNSesV6	All authoritative servers are responsive for the domain/[16]	`True`
nPrefixes(NSv4)	Number of unique BGP prefixes for NSes (IPv4/[17])	>=2
nPrefixes(NSv6)	Number of unique BGP prefixes for NSes (IPv6)/[17])	>=2
nAses(NSv4)	Number of unique ASes for NSes (IPv4) [18]	>=2
nASes(NSv6)	Number of unique ASes for NSes (IPv6) [18]	>=2
nGeoDiverseNSes	Number of NS distinct geographical locations [17]	>=2

Table 3.1: Critical Best Practices Metrics

3.1. Critical Best Practices

Table 3.1 summarizes the critical best practices for authoritative DNS servers operators. We define each practice as individual metrics, which will use in the second phase of this study – to measure them for the websites related to the Government of the Netherlands.

Next we expand of each individual metric and practice. For that, we use the example show in Figure 3.1, for a sample DNS zone: `example.nl`.

3.1.1. nNSes: number of NS records for a zone

Description: each domain name is required to have at least two authoriative DNS servers [15], *i.e.*, , two distinct NS records, in order to guarantee *some* level of redundancy, as having a single NS would be a single point of failure. In our example from Figure 3.1, this is shown by having two NS records: `ns1.example.nl` and `ns2.example.com`. Each NS record, in turn, may be ran by a different organization and using IP anycast, which provides extra redundancy (§3.2.5).

Reference: this best practice has been proposed on the original DNS standard [15], so we do not expect to find many domains names that do not follow it.

How to measure it: A `dig` command line tool equivalent of: `dig ns $domain_name`

3.1.2. nIP(NSv4): number of unique IP addresses for all NSes

Description: This metric consists in determing how many unique IPv4 addresses *host* the authoritative DNS servers. In our Figure 3.1, that would be the number of unique IPv4 addresses assocaited with both NS records (`ns1.example.nl` and `ns2.example.com`).

Notice that a single domain may have multiple NS records (fullfiling in this way, the §3.1.1). However, all of these NS records may have the *same* A records (for example, all pointing to 192.168.1.1, which would still create a single point of failure. Thus, the metric from §3.1.1, if analyzed alone, could provide a false sense of security.

Reference: This best practice is document on RFC2182 [17].

How to measure it: For each NS record, retrieve its A record(s) that must be publicly *routable*, *i.e.*, , valid and reachable IP address space. Then, count the number of unique records for all.

3.1.3. nIP(NSv6): number of unique IP addresses for all NSes

Description: Same as in §3.1.2, except it measures AAAA records (IPv6) instead of A records (IPv4).

3.1.4. ResponsiveNSesV4 :All authoritative servers are responsive for the domain

Description: In our example domain in Figure 3.1, a registrant (who owns the domain) sets two NS records for its domain (`ns1.example.nl` and `ns2.example.com`). However, these servers may not be active, may be not be authoritative for the zone in question (referred to as lame delegation [16]), and ultimately may not been able to provide authoritative information for the domain.

For example, if a user would ask data about Japan's DNS zone `.jp` to a `.nl` authoritative server (*e.g.*, : `dig ns example.jp @ns1.dns.nl`), the `.nl` would *refuse* to answer the question, indicating the NL server is not authoritative for `.jp`.

Reference: Lame delegations are defined in RFC1713 [16] and evaluated in [19].

How to measure it: This involve a series of steps.

1. Get the IP addresses of all NS records
2. For each address, send a SOA query or A or NS query about the domain name in question. If the response is OK (RCODE=0 [1]), then the server is properly configured. If not, then the server has an issue.

3.1.5. ResponsiveNSesV6 :All authoritative servers are responsive for the domain

Same as §3.1.4, except for IPv6 addresses.

3.1.6. nPrefixes(NSv4) Number of unique BGP prefixes for NSes (IPv4)

Description: IP addresses are announced on the Internet in blocks called "BGP prefixes" [20]. These prefix announcements contain information that help routers determine where address space can be reached. For example, suppose a telecom company announces a IP block address 192.168.0.0/24 (which covers 256 /32 addresses). This annoucement is received by neighboring routers, which propagate it even further.

For resilence, it is better to have, for a given DNS zone, *distinct* prefixes for all IP addresses of the NS records. This provides some isolation in case one of the route announcements experiences issues.

In our example in Figure 3.1, we see two addresses that *likely* belong to two different prefixes.

Reference: This falls in the category of having dissimilar infrastructure of authoritative servers. This is defined in [17].

How to measure it: For that, we have to analyze BGP prefix announcements in public sources of BGP data, such as RIPE RIS [21] and RouteViews [22]. To measure it, we must:

1. Get the IP addresses of all NS records
2. Determine which prefix announcements cover these addresses
3. Count the number of unique prefixes

3.1.7. nPrefixes(NSv6) Number of unique BGP prefixes for NSes (IPv4)

Same as §3.1.6, except for IPv6.

3.1.8. nAses(NSv4): Number of unique ASes for NSes (IPv4)

Description: As discussed in §3.1.6, IP addresses are announced in BGP using prefixes. This annoucement also contains what *Autonomous Systems*(ASes) are in the path to the prefix, and its *origin* AS. Ultimately, it's the origin AS that *hosts* the IP addresses in questions.

Metric	Description/Ref.	Value
nTLDs	Use more than one TLD for NS records/[18]	2
NS TTL	TTL values of NS records/[25, 26, 27]	>=3600s
A(NS) TTL	TTL values for A (NS) records[25, 26, 27]	>=1800s
AAAA(NS) TTL	TTL values for AAAA (NS) records[25, 26, 27]	>=1800s
nAnycastIPv4	Number of Anycast Auth Servers IPv4/[8]	>=1
nAnycastIPv6	Number of Anycast Auth Servers IPv6/[8]	>=1

Table 3.2: Recommended Best Practices Metrics

To improve resilience, it is recommended to have the IP addresses of your authoriative server in more than one AS, to avoid single points of failure if something goes wrong with a particular AS.

Reference: This falls in category of having dissimilar infrastructure of authoritative servers. This is defined in [17].

How to measure it: For that, we have to analyze BGP route announcements from public sources, such as RIPE RIS [21] and RouteViews [22].

1. Get the IP addresses of all NS records
2. Determine which announced prefixes cover the addresses
3. Determine what origin AS announces the BGP prefix
4. Count the number of unique origin ASes

3.1.9. nAses(NSv6): Number of unique ASes for NSes (IPv4)

Same as §3.1.8, except for IPv6 addresses.

3.1.10. nGeoDiverseNSes: Number of NS distinct geographical locations

Description: Authoritative nameservers shoud be placed in different geographical location in order to avoid that a physical disaster in a location (e.g. fire) can affect all the servers.

To improve resilience, it is recommended to put the nameservers in different cities.

Reference: RFC2182 [17] states that secondary servers should be at geographically distant locations.

How to measure it: For that, we have to analyze IP geolocation databases such as Maxmind [23] or run active measurements to identify locations using RIPE Atlas [24].

3.2. Recommended Best Practices

Table 3.2 summarized the *recommended* best practices. Recommended refers to practices that *improve* the dependability of the DNS, but not following them does not lead to a single point-of-failure, which, in turn would imply total unreachability.

Next we expand these recommended best practices.

3.2.1. nTLDs: number of unique TLDs used in the NS records

Description: this metric refers to the number of top-level domains (TLDs) used in the NS records. In the example of Figure 3.1, we see that the two NS records user different TLDs: `.com` and `.nl`. That means is one of these two TLDs would become unreachable, the `example.nl` zone could still be reachable via the `.nl` TLD.

Similarly, the criticial domain `digid.nl` has 4 NS records, from four different TLDs: `.nl`, `.eu`, `.org`, and `.com`.

Note that if resolvers do not already have NS records for `example.nl`, then the `.nl` authoritative servers must be reachable.

Another solution is to provide glue records for all the NS records, in that case the domain will use in-bailiwick records that will require only the `.nl` TLD to be reachable.

Reference: This practice has been long been known by the community, and is also documented by [18].

How to measure it: extract all NS records for a given domain, and count the distinct number of TLDs.

Caveat: note that many TLDs share the same DNS infrastructure, so one has to choose carefully which TLDs to host. For example, `.com` and `.net` use the same infrastructure, and are run by a single company (Verisign).

3.2.2. NS TTL value

Description: DNS record, such as the NS records in Figure 3.1, always have a time-to-live field (TTL), which tells DNS resolvers the maximum time the DNS responses should be kept in the DNS cache of the servers. DNS caches, as CR_n in Figure 2.1, are the cornerstone of DNS performance [25, 26, 27]: having a cached response drastically reduces the response time to clients. Moreover, in case of DDoS attacks, having *longer* TTLs (say minimum an hour) would allow clients behind resolvers with hot caches to *still be able to reach* the destination website, even though the DNS authoritative servers may be completely unreachable. Caching can therefore be seen as a *ephemeral* resilience.

Given these considerations, the proper choice for a TTL depends in part on multiple external factors – no single recommendation is appropriate for all scenarios. Organizations must weigh these trade- offs and find a good balance for their situation. Still, some guidelines can be reached when choosing TTLs:

- For general DNS zone owners, [27] recommends a longer TTL of at least one hour, and ideally 8, 12, or 24 hours. Assuming planned maintenance can be scheduled at least a day in advance, long TTLs have little cost and may, even, literally provide a cost savings.
- Users of DNS-based load balancing or DDoS-prevention services may require shorter TTLs: TTLs may even need to be as short as 5 minutes, although 15 minutes may provide sufficient agility for many operators. There is always a tussle between shorter TTLs providing more agility against all the benefits listed above for using longer TTLs.

Reference: We have previously investigated the role of caching in DDoS attacks in DNS in several studies [25, 26, 27].

How to measure it: To measure the TTL value of a record, one must obtain an authoritative answer by asking *direclty* the authoritative servers, and bypass local resolvers which may have a hot cache and decremented TTL values.

Caveat: There is some level of duplication in DNS: NS records can be found in both *parent* and *child* DNS zones. For example, the NS records for `example.nl` in Figure 3.1 can be found at the `.nl` authoritative servers (which are the "parent"), as well as in the "child" authoritative servers (`ns1.example.nl` and `ns2.example.com`). These values, however, may differ [28], given that these zones are typically managed by different organizations. However, most resolvers in the wild tend to follow the *child* authoritative server TTL [29]. For this reason, we will consider only the child TTL value.

3.2.3. A(NS) TTL

Description: in §3.2.2, we analyze the TTL of NS records for a given domain. These NS records, in turn, need to have A and/or AAAA addresses to be reachable – these are the IP addresses that are used to route packets. In Figure 3.1, that refers to the TTL value of the A records (192.168.1.1).

The TTLs for A/AAAA records should be shorter to or equal to the TTL for the corresponding NS records for in-bailiwick authoritative DNS servers, since [27] finds that once an NS record expires, their associated A/AAAA will also be re-queried when glue is required to be sent by the parents. For out-of-bailiwick servers, A, AAAA and NS records are usually all cached independently, so different TTLs can be used effectively if desired. In either case, short A and AAAA records may still be desired if DDoS-mitigation services are required.

Reference: We have previously investigated the role of caching in DDoS attacks in DNS in several studies [25, 26, 27].

How to measure it: To measure the TTL value of a record, one must obtain an authoritative answer by asking *directly* the authoritative server and bypassing local resolvers, which may have a hot cache and decremented TTL values.

Caveat NS records can be in or out of zone (in or out of bailiwick in DNS terminology). For example, the IP address of `ns1.example.nl` must be placed as a glue record in the parent DNS zone (`.nl`) for `example.nl`, given they share the same second-level domain (`example.nl`). This is different from `ns2.example.com`, which uses another TLD. In this case, the IP address (A record) is only available at the child authoritative server. (Most of zones, however, are out-of-bailick [28]). So we measure them accordingly to their setup.

3.2.4. AAAA(NS) TTL

Same as §3.2.3, except for AAAA (IPv6) records.

3.2.5. nAnycastIPv4: number of anycast-based authoritative server

Description IP anycast consists of annoucing the same IP prefixes from multiple locations [5]. Anycast is largely used in DNS [30], especially by operators of prominent authoritative servers. For example, all the root DNS servers use IP anycast.

IP anycast *fragments* the IP address space, and maps each fragment into a different anycast site. For example, in Figure 2.2, we see that `K-ROOT` has 75 anycast sites: the entire IPv4 is distributed among the 73 sites – which is done by BGP [8], where clients are mapped to nearby sites (nearby in BGP terms, and not necessarily geographical distance [31]). This distribution is not necessarily uniform, some sites may see far more clients than others.

In case of DDoS attacks against an authoritative server, we see that some sites experience the attack diferently [8]: some sites may remain up while others remain down. That behavior has been observed in the Root DNS Events on November 2015 [8]. As such, operators can, on-the-fly, configure their authoritative anycast DNS to try to steer DDoS traffic to one of few sites, while others may remain up.

Our goal is to determine which the A/AAAA addresses of the authoritative servers use anycast.

Reference: IP anycast is documented in [5]. Its DNS usage in [30]. Its relation to DDoS in [8]. And how to measure anycast in the wild is documented in [32, 33].

How to measure it: We will use the procedure described in [32] using the Anycast Testbed from SIDN. In short: we will use active measurements from an anycast network to measure the IP addresses from the government networks.

3.2.6. nAnycastIPv6: number of anycast-based authoritative server

Same as §3.2.5, except for AAAA (IPv6) records.

3.3. Immeasurable best practices

Our methodology can only account for metrics that can be measured on the IP layer (layer 3) and above. As such, any single-point-of-failure mitigation metric that is located *below* layer 3 is, in most cases, *immeasurable*. Although they are essential for the resilience of authoriative DNS servers, we consider them as out-of-scope in this study, given we cannot measure them.

Table 3.3 summarizes them. Next we detail each of them..

Metric	Description/Reference	Value
nPhysicalLocations	number of unique physical locations hosting the authoritative name servers	$>= 2$
nPhysicalLines	number of distinct physical lines connecting authoriative name servers	$>= 2$
nPhysicalServers	number of unique baremetal servers	$>= 2$
KeyServersOnClients	Place anycast sites or authoritative servers on key clients	NA

Table 3.3: Immeasurable Best Practices Metrics

3.3.1. nPhysicalLocations

Authoritative servers – either virtual or bare metal, should be placed in *distinct* physical locations, to avoid that any local related failures (attacks, power outages, etc.) affects the authoriative DNS servers altogether.

Consider the worst-case scenario, in which three authoritative servers are hosted in different IP address space, using different upstream providers, but all being physically hosted on the same, single datacenter: no matter how much redundancy is added, this setup still has a single point-of-failture, which is a single location.

As such, we recommend operators to use multiple physical locations to host their services.

3.3.2. nPhysicalLines

Similar to the number of physical servers, there must be multiple lines that connect authoriative servers to the Internet – not for each of them, but for all of the combined. The goal is to avoid a single point-of-failure.

3.3.3. nPhysicalServers

The last metric the number of physical servers hosting the authoriative DNS servers . One could run multiple authoritative DNS servers on a single bare metal server, ultimately removing redundancy. The goal of this metric is to avoid this.

3.3.4. KeyServersOnClients

For specific services, such as `digid.nl`, it may be worth to add *anycast sites* of authoritative servers on key client networks – for example, the networks of major ISPs and where most clients come from.

Depending on the type of attack, this setup may provide DNS services to clients while other parts of the network may be under attack. For example, suppose a particular DDoS attacks the networks on a IXP. Clients can still be able to resolve the domain if they have access to servers on their ISP's network. This pratice only improve resilience in cases the client's network are not able to reach the networks of the authoritative servers.

In addition to that, we intend to write a *speculative* scenario, in which The Netherlands is "disconnected" by some reason (DDoS attack, for example), from the global Internet. In this scenario, we will estimate how much of the domain names related to the government will still be able to be resolved.

4

Next steps

The metrics discussed here will be implemented in our tooling to measure resilience of the DNS of the Netherlands government. We will first draft a measurement plan, and then share it with our colleagues at the government.

Bibliography

[1] P. Mockapetris, "Domain names - concepts and facilities," IETF, RFC 1034, Nov. 1987. [Online]. Available: http://tools.ietf.org/rfc/rfc1034.txt

[2] M. Antonakakis, T. April, M. Bailey, M. Bernhard, E. Bursztein, J. Cochran, Z. Durumeric, J. A. Halderman, L. Invernizzi, M. Kallitsis, D. Kumar, C. Lever, Z. Ma, J. Mason, D. Menscher, C. Seaman, N. Sullivan, K. Thomas, and Y. Zhou, "Understanding the Mirai botnet," in *Proceedings of the 26th USENIX Security Symposium*. Vancouver, BC, Canada: USENIX, Aug. 2017, pp. 1093–1110. [Online]. Available: https://www.usenix.org/system/files/conference/usenixsecurity17/sec17-antonakakis.pdf

[3] N. Perlroth, "Hackers used new weapons to disrupt major websites across U.S." *New York Times*, p. A1, Oct. 22 2016. [Online]. Available: http://www.nytimes.com/2016/10/22/business/internet-problems-attack.html

[4] P. Hoffman, A. Sullivan, and K. Fujiwara, "DNS Terminology," IETF, RFC 8499, Nov. 2018. [Online]. Available: http://tools.ietf.org/rfc/rfc8499.txt

[5] C. Partridge, T. Mendez, and W. Milliken, "Host Anycasting Service," IETF, RFC 1546, Nov. 1993. [Online]. Available: http://tools.ietf.org/rfc/rfc1546.txt

[6] J. Abley and K. Lindqvist, "Operation of Anycast Services," IETF, RFC 4786, Dec. 2006. [Online]. Available: http://tools.ietf.org/rfc/rfc4786.txt

[7] Root Server Operators, "Root DNS," May 2020, http://root-servers.org/.

[8] G. C. M. Moura, R. de O. Schmidt, J. Heidemann, W. B. de Vries, M. Müller, L. Wei, and C. Hesselman, "Anycast vs. DDoS: Evaluating the November 2015 root DNS event," in *Proceedings of the ACM Internet Measurement Conference*. Santa Monica, California, USA: ACM, Nov. 2016, pp. 255–270. [Online]. Available: https://www.isi.edu/%7ejohnh/PAPERS/Moura16b.html

[9] Google, "Public DNS," https://developers.google.com/speed/public-dns/, Nov. 2020. [Online]. Available: https://developers.google.com/speed/public-dns/

[10] OpenDNS, "Setup Guide: OpenDNS," https://www.opendns.com/, Mar. 2021. [Online]. Available: https://www.opendns.com/

[11] Quad9, "Quad9 | Internet Security & Privacy In a Few Easy Steps," https://quad9.net, Jan. 2018.

[12] 1.1.1.1, "The Internet's Fastest, Privacy-First DNS Resolver," https://1.1.1.1/, Apr. 2018. [Online]. Available: https://1.1.1.1/

[13] J. Scudder, R. Fernando, and S. Stuart, "BGP Monitoring Protocol (BMP)," IETF, RFC 7854, Jun. 2016. [Online]. Available: http://tools.ietf.org/rfc/rfc7854.txt

[14] R. Arends, R. Austein, M. Larson, D. Massey, and S. Rose, "Protocol Modifications for the DNS Security Extensions," IETF, RFC 4035, Mar. 2005. [Online]. Available: http://tools.ietf.org/rfc/rfc4035.txt

[15] P. Mockapetris, "Domain names - implementation and specification," IETF, RFC 1035, Nov. 1987. [Online]. Available: http://tools.ietf.org/rfc/rfc1035.txt

[16] A. Romao, "Tools for DNS debugging," IETF, RFC 1713, Nov. 1994. [Online]. Available: http://tools.ietf.org/rfc/rfc1713.txt

[17] R. Elz, R. Bush, S. Bradner, and M. Patton, "Selection and Operation of Secondary DNS Servers," IETF, RFC 2182, Jul. 1997. [Online]. Available: http://tools.ietf.org/rfc/rfc2182.txt

[18] M. Allman, "Comments on DNS Robustness," in *Proceedings of the Internet Measurement Conference 2018*, ser. IMC '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 84–90. [Online]. Available: https://doi.org/10.1145/3278532.3278541

[19] G. Akiwate, M. Jonker, R. Sommese, I. Foster, G. M. Voelker, S. Savage, and K. Claffy, "Unresolved issues: Prevalence, persistence, and perils of lame delegations," in *Proceedings of the ACM Internet Measurement Conference*, ser. IMC '20. New York, NY, USA: Association for Computing Machinery, 2020, p. 281–294.

[20] Y. Rekhter, T. Li, and S. Hares, "A Border Gateway Protocol 4 (BGP-4)," IETF, RFC 4271, Jan. 2006. [Online]. Available: http://tools.ietf.org/rfc/rfc4271.txt

[21] Reseaux IP Europeens Network Coordination Centre (RIPE NCC), "Routing Information Service (RIS)," 2021. [Online]. Available: http://www.ripe.net/data-tools/stats/ris

[22] University of Oregon , "Route Views Project," 2021. [Online]. Available: http://www.routeviews.org/

[23] Maxmind, "Maxmind," 2012. [Online]. Available: http://www.maxmind.com/

[24] RIPE Network Coordination Centre, "RIPE Atlas," https://atlas.ripe.net, 2020.

[25] G. C. M. Moura, W. Hardaker, J. Heidemann, and M. Davids, "Considerations for Large Authoritative DNS Servers Operators," Internet Engineering Task Force, Internet-Draft draft-moura-dnsop-authoritative-recommendations-09, Aug. 2021, work in Progress. [Online]. Available: https://datatracker.ietf.org/doc/html/draft-moura-dnsop-authoritative-recommendations-09

[26] G. C. M. Moura, J. Heidemann, M. Müller, R. de O. Schmidt, and M. Davids, "When the dike breaks: Dissecting DNS defenses during DDoS," in *Proceedings of the ACM Internet Measurement Conference*. Boston, MA, USA: ACM, Oct. 2018, pp. 8–21. [Online]. Available: https://www.isi.edu/%7ejohnh/PAPERS/Moura18b.html

[27] G. C. M. Moura, J. Heidemann, R. de O. Schmidt, and W. Hardaker, "Cache me if you can: Effects of DNS Time-to-Live," in *Proceedings of the ACM Internet Measurement Conference*. Amsterdam, the Netherlands: ACM, Oct. 2019, pp. 101–115. [Online]. Available: https://www.isi.edu/%7ejohnh/PAPERS/Moura19b.html

[28] R. Sommese, G. Moura, M. Jonker, R. van Rijswijk-Deij, A. Dainotti, k. claffy, and A. Sperotto, "When parents and children disagree: Diving into DNS delegation inconsistency," in *Passive and Active Measurement Conference (PAM)*, 2020-03.

[29] G. C. M. Moura, S. Castro, W. Hardaker, M. Wullink, and C. Hesselman, "Clouding up the Internet: How Centralized is DNS Traffic Becoming?" in *Proceedings of the ACM Internet Measurement Conference*, ser. IMC '20. New York, NY, USA: Association for Computing Machinery, 2020, p. 42–49.

[30] D. McPherson, D. Oran, D. Thaler, and E. Osterweil, "Architectural Considerations of IP Anycast," IETF, RFC 7094, Jan. 2014. [Online]. Available: http://tools.ietf.org/rfc/rfc7094.txt

[31] R. d. O. Schmidt, J. Heidemann, and J. H. Kuipers, "Anycast latency: How many sites are enough?" in *Proceedings of the Passive and Active Measurement Conference*. Sydney, Australia: Springer, Mar. 2017, pp. 188–200. [Online]. Available: http://www.isi.edu/%7ejohnh/PAPERS/Schmidt17a.html

[32] R. Sommese, L. Bertholdo, G. Akiwate, M. Jonker, van Rijswijk-Deij, Roland, A. Dainotti, K. Claffy, and A. Sperotto, "MAnycast2—using anycast to measure anycast," in *Proceedings of the ACM Internet Measurement Conference*. Pittsburgh, PA, USA: ACM, Oct. 2020.

[33] R. Sommese, G. Akiwate, M. Jonker, G. C. Moura, M. Davids, R. van Rijswijk-Deij, G. M. Voelker, S. Savage, K. Claffy, and A. Sperotto, "Characterization of Anycast Adoption in the DNS Authoritative Infrastructure," Mar. 2020.

Authors

dr. Giovane C. M. Moura is a Data Scientist with SIDN Labs, the research arm of SIDN, the Netherland's .nl top-level domain operator. His research focus on bringing academic rigor to network operations, to improve performance, security and stability of networked systems. I am also a research guest at TU Delft's CyberSecurity group. He obtained his Ph.D. in 2013 from the University of Twente.

dr.ir. Mattijs Jonker is assistant professor and research scientist at University of Twente. His research is on network security in a broad sense and involves extensive data science and Internet measurement. He is one of two architects of the award-winning OpenINTEL project, which measures sizable parts of the domain name system for security research. Mattijs earned his Ph.D. at the University of Twente in 2019, cum laude, and also holds a MSc specialization in Cyber Security.

Raffaele Sommese, MSc is a PhD candidate at the University of Twente. His research focuses on analyzing and characterizing DNS vulnerabilities and misconfiguration in order to improve protection against and the prevention of DNS DDoS attacks. Raffaele received his Master's degree in Computer Engineering from Politecnico di Torino in 2018.