Standardizing Confidentiality Protections for Domain Name System (DNS) Exchanges: Multiple Approaches, New Functionality

Burton S. Kaliski Jr. | VERISIGN

The article's Digital Object Identifier (DOI) is: 10.1109/MCOMSTD.201.2000085

CONTENTS

This article was originally published as B. S. Kaliski, "Standardizing Confidentiality Protections for Domain Name System Exchanges: Multiple Approaches, New Functionality," in IEEE Communications Standards Magazine, vol. 5, no. 3, pp. 26-32, September 2021, doi: 10.1109/MCOMSTD.201.2000085.

ABSTRACT

Confidentiality protections have become a major focus of standards development for the Domain Name System (DNS) protocol. DNS encryption techniques as well as alternative techniques with lower operational impact have both emerged. This article provides a high-level overview of these techniques and the considerations for applying them in various parts of the DNS ecosystem. The article also discusses how the standardization of DNS encryption can facilitate a new multi-resolver architecture where clients route queries to one or more special-purpose resolvers associated with enterprise or application namespaces.

INTRODUCTION

The Domain Name System (DNS) is the fundamental building block for navigating from names to resources on the internet, emerging from an era when interconnection rather than information security was the primary motivation. Since its inception in 1983, the DNS has gradually improved its security features, as well as its navigational and security capabilities.

Over a decade ago, the standards organization responsible for the DNS protocol, the Internet Engineering Task Force (IETF), introduced the DNS Security Extensions (DNSSEC), a set of specifications for enhancing the integrity of DNS exchanges. The IETF has recently turned its attention to standards for enhancing the confidentiality of these exchanges, through the formation of the DNS Private Exchange (DPRIVE) working group.

Just as standards developers found a way to balance data integrity objectives with the operational realities of an internet infrastructure protocol when designing DNSSEC, so too standards developers are now balancing data confidentiality objectives with operational concerns. The balance is reflected in a variety of standards efforts including both DNS encryption protocols and alternative techniques that reduce the sensitivity of the information on the various DNS protocol exchanges.

Standards make it easier for communicating parties to interoperate when fulfilling a previously agreed function. They also open up opportunities to pursue new functionality. The layering of abstraction upon abstraction has been a hallmark of the development of internet applications for decades. Once one set of services is broadly in place, further enhancements can readily be built on that foundation.

This pattern is starting to play out with DNS standards related to confidentiality protection. On the one hand, there's significant effort already underway to specify DNS encryption protocols and alternative techniques that can help meet confidentiality objectives in various parts of the DNS ecosystem. The first half of this article will review these efforts and suggest a way of fitting the different approaches together. On the other, with these protections in place, there's also an opportunity for DNS to provide new functionality. This functionality is the focus of the second half of this article.

The purpose of this article is then two-fold: to give an overview of the techniques currently being considered for standardization; and to provide a preview of future areas of standardization that could build on the current efforts.

STANDARDIZATION EFFORTS

The industry standards related to DNS, DNSSEC and the confidentiality protection techniques discussed in this article are within the purview of the IETF. Within the IETF, most of the relevant work is taking place in one of three working groups: DPRIVE, mentioned above; Domain Name System Operations (DNSOP); and Adaptive DNS Discovery (ADD). (The author's employer actively participates in these working groups and his colleagues are listed as authors on several of the standards documents mentioned in this article.)

Information about the IETF's standards activities can be found on ietf.org. The specifications resulting from these activities, called Requests for Comments (RFCs), are published on rfc-editor.org. Some RFCs are *standards track* in the sense of being required for, or intended to be required for, interoperability. Others are *experimental* or *informational*, for reference by the internet community. The IETF is a voluntary standards organization.

The article assumes that domain names are managed within the traditional, global DNS under the Internet Assigned Number Authority's (IANA) root zone. Decentralized name systems have recently been proposed as alternatives to the global DNS. Some may employ the IETF-standardized protocols discussed here, but others are based on blockchain and other techniques that are currently not within the scope of DNS standards and are not discussed further in this article.

DNS RESOLUTION

The story of DNS begins with the usual occurrence that happens millions of times a second around the world: a client asks a DNS resolver a query like "What is www.example.TLD's Internet Protocol (IP) address?"

The resolution process follows a chain of *delegation of authority* referrals that enable the resolver to determine which of the servers is ultimately *authoritative* for DNS records about the domain name of interest, e.g., www.example.TLD and its associated IP address, conveyed in a DNS A record. The root servers, top-level domain (TLD) servers, second-level domain (SLD) servers and all the rest of the servers in the chain are collectively referred to as the *authoritative name servers* in the DNS resolution ecosystem.

Figure 1 shows the DNS resolution process, following an emerging practice called query name minimization (see below). The process includes the following steps:

1. The client asks the resolver for a DNS record (e.g., the IP address) associated with www.example.TLD;
2. The resolver asks a root server for a referral to a TLD server for .TLD;
3. The root server returns a referral;
4. The resolver asks a TLD server for a referral to an SLD server for example.TLD;
5. The TLD server returns a referral;
6. The resolver asks the SLD server for the DNS record associated with www.example.TLD;
7. The SLD server returns the DNS record; and
8. The resolver returns the record to the client.

Note that some or all of the intermediate steps may be skipped if resolver already has an answer in its cache as a result of previous processing.

There are approximately 363.5 million domain names registered as of Mar. 31, 2021 [1]. The global DNS includes more than 1,500 TLDs [2]. DNS resolution processing runs continuously, behind the scenes, for all these domains and the subdomains under them such as www.example.com. DNS supports all kinds of internet transactions and applications that interact with named resources — including the connections to the websites for this magazine, www.comsoc.org and ieeexplore.ieee.org. (The author's employer operates two of the 13 root servers as well as the .com and .net TLD servers, among others.)

From a historical perspective, the resolution process hasn't always been as shown in Figure 1. Indeed, there's been one subtle but significant change in the past few years, one that is now gaining rapid adoption.

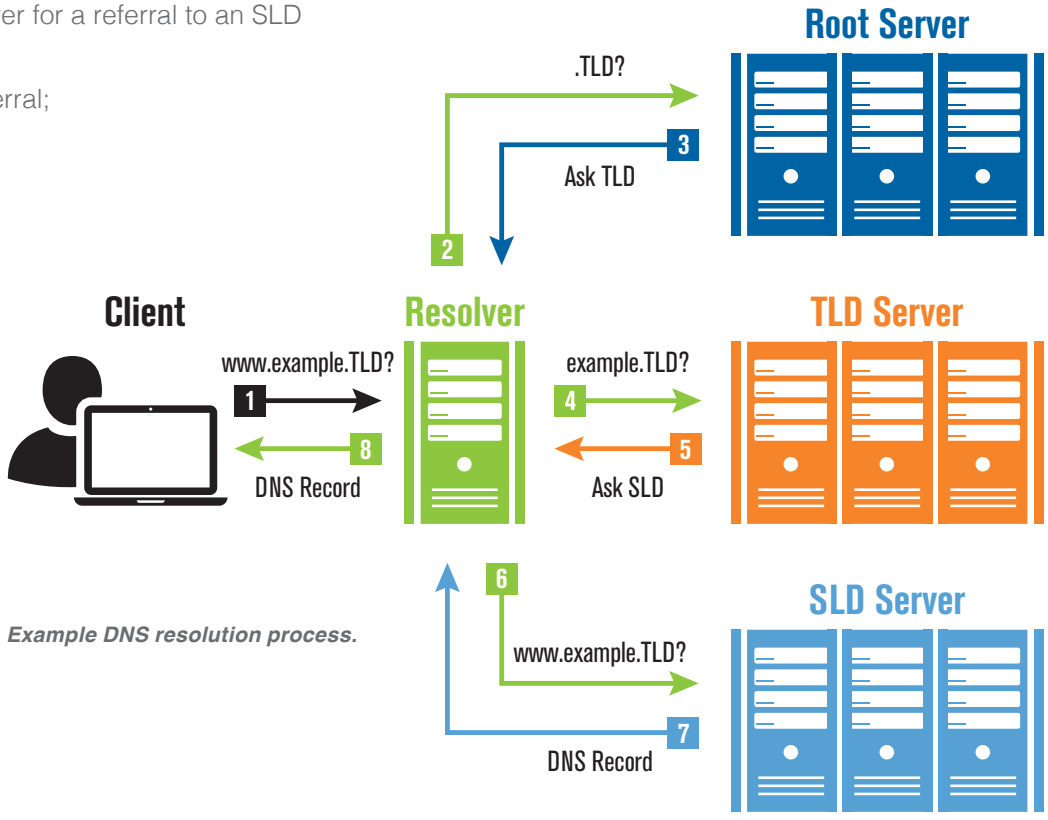

Figure 1. Example DNS resolution process.

QNAME MINIMIZATION

In the early days of DNS, it wasn't obvious that the root and TLD servers would always answer with a referral. Indeed, in principle, they might actually have known the answer themselves, rather than delegating authority to another server at a lower level. But today, the policies governing these servers have made them *delegation only* for practical purposes. Under policies overseen by the Internet Corporation for Assigned Names and Numbers (ICANN), the management of each TLD and the names below it is delegated to a TLD registry operator. Each TLD registry operator, in turn, delegates the authority to manage SLDs and the names below them to registrants.

When the root server receives a query about a domain name at any level below the root, it will generally only answer authoritatively if it the TLD in the domain name *doesn't* exist. Otherwise, it will answer with a referral to the TLD's name server.

Similarly, a TLD server will generally answer a query about a domain name at a level below the TLD with a referral to an SLD server, except when the SLD doesn't exist.

With this understanding, sending the full domain name of interest, like www.example.com, to a root or TLD server, is more information than necessary. The resolver need only send as much information as the name server requires to make the referral, i.e., it should just disclose one label, the TLD, to the root server, and just two labels, the SLD and the TLD, to the TLD server.

Query name minimization or *qname minimization* [3] is one technique for reducing the disclosure of information to the name servers.[1] A resolver implementing qname minimization adapts the query it sends based on the resolver's knowledge of the DNS delegation structure, arriving at a process like the one described in Figure 1.

A resolver can also adapt the query type that it sends to the name server for additional reduction in the information disclosed. The referral will be the same regardless which query type the resolver sends, so the resolver can just send a constant type to the root and TLD servers, such as a request for an A record, regardless of the type the client is interested in.

Measurements taken at the .com and .net TLD servers operated by Verisign indicate that in Feb. 2021, 55% of all queries received by these servers had exactly two labels, whereas in Jan. 2018, only 30% of queries had two labels. The fraction of queries with three labels correspondingly decreased. The change over this three-year period is likely attributable to the deployment of qname minimization by many resolvers. Independent measurement research in 2019 also reports a "slow but steady adoption" and provides a discussion of benefits and risks [4].

Third-level registrations. Some TLDs reserve certain SLDs for registration purposes and delegate authority for third-level names under those SLDs to registrants. For instance, the domain name for the China Computer Federation, a sister society of the IEEE Computer Society, is ccf.org.cn. The TLD server for .cn refers queries for this domain name to the third-level domain server for ccf.org.cn. For simplicity, this article will assume that TLD registry operators delegate authority just at the SLD level, but the techniques described, including qname minimization, and the standards related to them can accommodate the third-level case as well.

The Public Suffix List, maintained by Mozilla, is the de facto standard for indicating the delegation points where second-level, third-level and other registrations may occur. See **publicsuffix.org** for additional information.

1. In 2015, Verisign announced a royalty-free license to its qname minimization patents in connection with IETF standardization efforts associated with RFC 7816. See IETF IPR disclosure 2542.

HOW TO PROTECT, NOT WHETHER TO ENCRYPT

Over the past several years, questions about how to protect information exchanged during DNS resolution have come to the forefront. One of these questions was posed first to DNS resolver operators in the middle of the last decade, and is now being brought to authoritative name server operators: “to encrypt or not to encrypt?”

In any system, rather than asking *whether* to deploy encryption for a particular exchange, a more important question is *how* to address the information protection objectives for that exchange — whether by encryption, alternative techniques, or some combination thereof.

Encryption can improve confidentiality and integrity by making it harder for an adversary to view or change data. However, encryption can also impair availability by making it easier for an adversary to exhaust a server’s resources, including network bandwidth, memory, and computation, which can then prevent legitimate users from obtaining service.

Resource exhaustion and denial of service attacks are especially a concern for essential internet infrastructure protocols like DNS that are expected always to be available to applications. This expectation motivates the implementation of high-availability designs, operational processes, and attack mitigation mechanisms.

Performance and operational risk were taken into account by standards developers in the design of DNSSEC, the last major upgrade to the DNS protocol. DNSSEC adopted an approach where the full set of positive and negative answers to potential queries could be generated offline in advance of the actual queries. The design decision avoided the need for name servers either to store cryptographic keys or to perform cryptographic operations in real time and has helped name servers that support DNSSEC —including the root servers and nearly all TLD servers — to maintain high availability today.

DNS encryption standards should similarly be designed in such a way that they can be deployed in high-availability applications. Measurement studies that analyze the performance of proposed protocols under large-scale attacks will also be important, in addition to those that assess response times for ordinary traffic (see, e.g., [5], for a recent study on the latter aspect). In addition, to cover the cases where encryption is not deployed, or is otherwise not available, standards developers should also develop other techniques for improving confidentiality protection that have lower operational risk.

Several important DNS confidentiality enhancements have emerged from recent IETF activities. The techniques are summarized in Table 1 and are described in the next two sections.

ENCRYPTION TECHNIQUES

DNS has traditionally been run over the UDP and TCP protocols, which are unencrypted. One way to enhance the confidentiality of DNS exchanges is to run the DNS protocol over an encrypted transport. So far, the IETF has specified two standards-track techniques for doing so:

Technique	Description	Reference	Status	Focus
Encryption Techniques				
DNS-over-TLS (DoT)	Parties exchange DNS traffic over TLS	RFC 7858	Proposed standard	Client-to-resolver; extension to resolver-to-authoritative underway
DNS-over-HTTPS (DoH)	Parties exchange DNS traffic over HTTPS	RFC 8484	Proposed standard	Client-to-resolver, especially from applications such as browsers
Minimization Techniques				
Qname Minimization	Resolver reduces amount of information in queries to just enough to get referral to next level of DNS hierarchy	RFC 7816	Experimental; standards-track in development	Resolver-to-authoritative, especially at root and TLD levels
NXDOMAIN Cut Processing	Resolver broadens interpretation of NXDOMAIN response to conclude that subdomains of queried domain don't exist either, thereby avoiding further queries	RFC 8020	Proposed standard	Resolver-to-authoritative at all levels
Aggressive DNSSEC Caching	Resolver broadens interpretation of negative DNSSEC response to conclude that additional domains specified in negative response range don't exist, thus avoiding further queries	RFC 8198	Proposed standard	Resolver-to-authoritative at all levels
Hyperlocal Root	Resolver operates with local copy of root zone file, avoiding root server queries entirely	RFC 8806	Informational	Resolver-to-root

Table 1. Summary of DNS confidentiality protection techniques discussed in this article.

- **DNS-over-TLS or DoT** [6], which runs DNS over the Transport Layer Security (TLS) protocol. The RFC describing this technique is a proposed standard. The specification focuses on the client-to-resolver exchange, and standards development is underway to extend it to the resolver-to-authoritative exchanges.
- **DNS-over-HTTPS or DoH** [7], which runs over Hypertext Transport Protocol Secure (HTTPS). The RFC describing this technique is also a proposed standard. The specification again focuses on the client-to-resolver exchange. DoH is particularly well matched to the case when the client is an application, such as a web browser, that is already running HTTPS for other purposes. DNS queries can then be interleaved with application requests over the same connection to an application server that also takes the role of a name server.

In addition, there is a proposal to run DNS over QUIC, a new encrypted transport protocol currently in standards development. DNS over QUIC or DoQ is under consideration by the DPRIVE working group.

MINIMIZATION TECHNIQUES

Another way to enhance confidentiality is with techniques that reduce the sensitivity of information on the traditional resolver-to-authoritative exchange. As a category, such techniques can be called *minimization techniques*.

Minimization techniques can achieve their goal either by putting less information into queries or by taking more information out of responses, thereby reducing the need for subsequent queries.

The IETF has specified four minimization techniques so far:

- **Qname minimization** [3], described above, puts less information into queries to the name server system. The RFC describing this technique has experimental status. Its successor, currently in development, is intended for the standards track.
- **NXDOMAIN cut processing** [8] takes more information out of negative responses from the name server system. NXDOMAIN is the error code a requester receives when a domain name doesn't exist. For historical reasons, some name servers also return NXDOMAIN when a domain name *does* exist and has subdomains but doesn't have DNS records itself. Because of the ambiguity, resolvers have traditionally not drawn conclusions about the status of subdomains based on an NXDOMAIN response for a domain. NXDOMAIN cut processing resolves the ambiguity, confirming, as the title of the referenced RFC says, that "there really is nothing underneath." The resolver can then answer queries for any of the subdomains on its own, reducing the need for subsequent queries to the name servers. The RFC describing NXDOMAIN cut processing is a proposed standard.
- **Aggressive DNSSEC caching** [9] takes more information out of negative responses when DNSSEC is involved. A typical negative response in DNSSEC references two domain names that *do* exist, with the implication that no other domain names exist between them in some defined ordering. A resolver implementing aggressive DNSSEC caching answers queries on its own for all of the domains in the range between the two endpoints, not just the domain name that it received the negative response for. The broader interpretation reduces the need for further queries to the name servers. The RFC describing this technique is a proposed standard.
- **Hyperlocal root** [10] takes the maximum amount of information from a root server's response: the resolver requests a copy of the full root zone file. A resolver implementing this technique can then answer queries about records in the root zone on its own. With a private "loopback" version available, a resolver doesn't send queries on the traditional resolver-to-root exchange at all. However, the resolver will need to ensure that its copy of the root zone file remains synchronized with the authoritative one served by the 13 root servers. The RFC describing the hyperlocal root technique is on the informational track. A supporting technique, the ZONEMD record [11], provides a way to authenticate a full zone file. The RFC describing this record is a proposed standard. Another supporting technique is the authoritative DNS zone transfer protocol (AXFR) [12], which provides a way for the resolver to obtain a copy of a zone file. Its RFC is also a proposed standard. An upgrade to AXFR that includes encryption, XFR over TLS (XoT), is currently in standards development in the DPRIVE working group.

THREE EXCHANGES, THREE APPROACHES

The DNS resolution ecosystem components involved in the resolution process in Figure 1 can be considered as consisting of three types of exchanges:

- Resolver-to- root and resolver-to-TLD — the navigational levels;
- Resolver-to-SLD and below; and
- Client-to-resolver.

Each exchange brings a different set of considerations for standardizing and deploying DNS confidentiality protections.

1. Resolver-to-Root and Resolver-to-TLD

The resolver-to-authoritative exchange at the root level enables DNS resolution for all underlying domain names; the exchange at the TLD level does the same for all names under a TLD. These exchanges provide *global navigation* for all names, benefiting all resolvers and therefore all clients, and making the availability objective paramount.

As a resolver generally services many clients, information exchanged at these levels represents aggregate interests in domain names, not the direct interests of specific clients. The sensitivity of this aggregated information is therefore relatively low compared to information on the resolver-to-client exchange. However, the full domain name of interest to a client has conventionally been sent to servers at the root and TLD levels, even though this is more information than they need to know to refer the resolver to authoritative name servers at lower levels of the DNS hierarchy.

Qname minimization reduces the information exchanged to just the resolver's aggregate interests in TLDs and SLDs. NXDOMAIN cut processing can make qname minimization more effective with non-existent domain names, and aggressive DNSSEC caching can reduce the amount of information exchanged even further, with the resolver handling requests for some non-existent TLDs and SLDs on its own. Resolvers also have the option of running a local copy of the root zone. With one or more such minimization techniques in place, root and TLD operators as well as resolver operators can then weigh the benefit of the further protection offered by DNS encryption against the operational risk of a protocol change affecting both sides of the exchange. The operators of the 13 root servers recently issued a joint statement expressing similar considerations [13].

Whether or not encryption is deployed, minimization techniques can still be valuable. Encryption protects against disclosure to or modification by outside parties but not against disclosure to (or by) the name server itself. Even though the sensitivity of the information on these exchanges may be relatively low for reasons noted above, without minimization, it's still more than the name server needs to know.

The standards efforts most relevant to this exchange are the specifications for the various minimization techniques. If a name server does support DNS encryption, however, another standard will be needed: a way for a name server to indicate to a resolver that it supports encryption. For simplicity and backwards compatibility, it will be important that if a name server doesn't support encryption, it won't have to do anything differently than it already does.

2. Resolver-to-SLD and Below

The resolver-to-authoritative exchanges at the SLD level and below enable DNS resolution within specific namespaces. These exchanges provide *local optimization*, benefiting all resolvers and all clients interacting with the included namespaces.

The information exchanged at these levels represents the aggregate interests of the resolver's clients. When a resolver is requesting a geographically optimized response, the information may also include client-related information such as a client's subnet, as outlined in RFC 7871 [14].

Qname minimization may be applied at the SLD level and below if the full domain name has more than three labels. NXDOMAIN cut processing and aggressive DNSSEC caching may also be applicable. However, minimization techniques may not help as much at the lowest-level name server involved as they do at the higher levels because the lowest-level name server will generally need to see the full domain name that the resolver is interested in.

The specifications for DNS encryption are relevant here if a name server chooses to support DNS encryption. As discussed above, such a name server will need a way to indicate to a resolver that it supports encryption. Given that there are already multiple DNS encryption protocols, the name server will also need a way to indicate which protocols it supports and any configuration parameters for those protocols.

Another interesting design question for standards developers is how to handle the case where the resolver is able to establish an encrypted session with the name server but isn't able to authenticate the name server's identity. As of this writing, this design question is still open, with several proposals under discussion.

3. Client-to-Resolver

The client-to-resolver exchange enables navigation to all domain names for all clients of the resolver.

The information exchanged here represents the interests of each specific client. The sensitivity of this information is therefore relatively high, making confidentiality vital. Minimization techniques don't apply here because the resolver needs the full domain name and the client-specific information included in the request.

The specifications for DNS encryption are most relevant here. Development is also underway on a standard way for a client to determine whether a resolver supports DNS encryption, or to discover another resolver that does (a *designated resolver* in the initial ADD working group terminology).

Designating a specific set of resolvers for a network or enterprise environment can help the network operator or enterprise detect and protect against command-and-control, data exfiltration and other attacks that might be facilitated through direct, encrypted connections to arbitrary external DNS resolvers [15].

Note that some clients use security and confidentiality solutions at different layers of the protocol stack (e.g., a virtual private network (VPN)), which could be applied to protect DNS exchanges, rather than the DNS-specific encryption techniques.

NEW FUNCTIONALITY: A MULTI-RESOLVER ARCHITECTURE

As with the introduction of other standards, one may expect that new features will emerge once standards for DNS encryption are in place. There are at least three reasons:

1. DNS encryption will require major upgrades to software implementations across the DNS ecosystem, an infrequent occurrence in the nearly four-decade history of the protocol. This reopening of the DNS software stack makes it a good time to consider related changes to the feature set.
2. The main security protocol underlying the DNS encryption techniques currently being considered for standardization, TLS, supports three features that new services can build on for additional purposes: server authentication, transport encryption and client authentication.
3. DNS encryption makes it possible to interact securely with enterprise- and application-specific resolvers for resolution of names in their own namespaces. Such special-purpose resolvers can offer clients enhanced DNS resolution for their namespaces without impacting a client's DNS resolution for other namespaces.

Recall that in the conventional DNS resolution process described in Figure 1, the client sends all its queries to a single, general-purpose resolver. The resolver therefore learns its clients' full interests in domain names.

Special-purpose resolvers enable a new *multi-resolver architecture*. As shown in Figure 2, the client routes its DNS queries to one of several resolvers. If the query is for a domain name is in a namespace associated with a special-purpose resolver, SPR-1 Namespace through SPR-n Namespace, then the client sends the query to the designated special-purpose resolver for the namespace, SPR-1 through SPR-n. Otherwise, the client sends the query to a general-purpose resolver, GPR. A special-purpose resolver only learns its clients' interests in names in the resolver's designated namespace, not their interests in names in other namespaces.

The operator of a special-purpose resolver can be the same as the operator of the resources in the designated namespace. The client's interaction with the special-purpose resolver can therefore stay within the need-to-know guidance on information disclosure. As a result, special-purpose resolvers can help resource operators optimize clients' interaction with resources in a namespace. It is also worth noting that the commonality of name server and resource operators was one consideration in standardizing DoH. In particular, a client can interleave DNS queries and application requests over a single HTTPS connection to the same entity's servers, thereby concealing whether it's doing a DNS lookup or an application transaction.

The multi-resolver architecture may itself be considered as a DNS confidentiality protection technique. Indeed, the architecture combines both types of confidentiality protection discussed in this article. First, the client's

exchange with the resolver is protected by DNS encryption. Second, the information sent over this exchange is limited to the resolver's designated namespace, a form of minimization. In addition, depending on the implementation, the special-purpose resolver can potentially be provisioned with the zone files for some parts of its designated namespace, thereby avoiding further resolver-to-authoritative queries entirely for those parts. For other parts, the confidentiality protection techniques described above for general-purpose resolvers could be applied to the special purpose resolver's exchanges with authoritative name servers.

Standards development is underway in the ADD working group on techniques supporting this architecture, as another aspect of the designated resolver concept mentioned above. The designation in this case is that the resolver is associated with a specific namespace. The author has recently described two examples of enhanced DNS resolution functionality that could be built on this architecture [16].

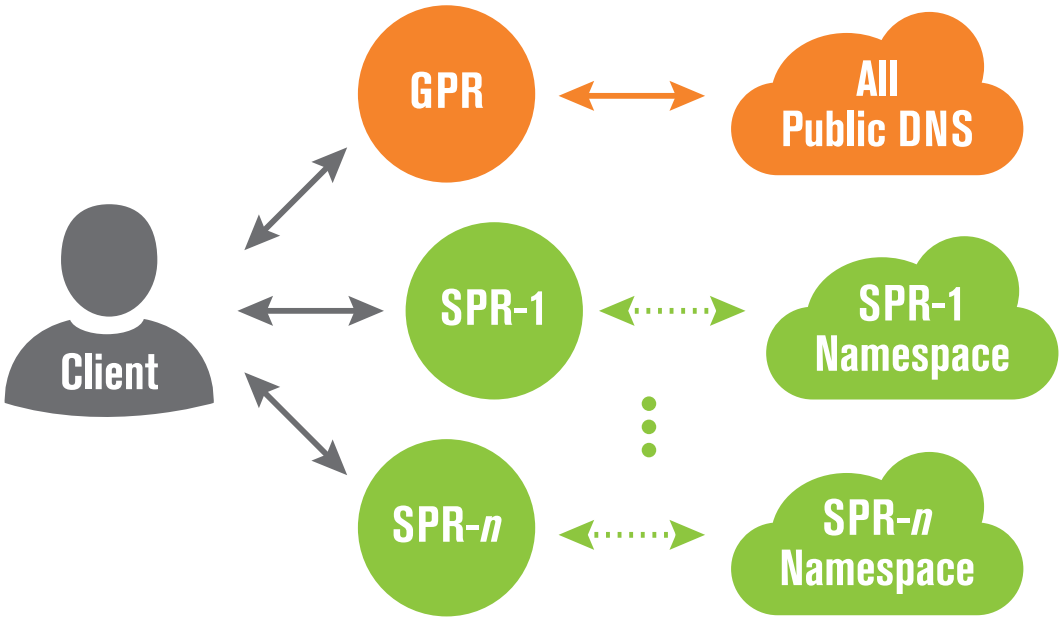

Figure 2. Multi-resolver architecture. GPR = general-purpose resolver, SPR = special-purpose resolver.

CONCLUSION

DNS encryption standards are bringing the historically unencrypted DNS protocol into a modern mode with cryptographic confidentiality protections. Standards developers have also specified alternative techniques that reduce the risk of disclosure of sensitive information with lower operational risk. The application of DNS encryption and alternatives reflects the reality of DNS as an essential internet infrastructure protocol and the classic information protection balance among confidentiality, integrity and availability.

The multi-resolver architecture facilitated by DNS encryption continues the modernization of DNS. In addition to reducing the amount of information disclosed to a given resolver, the architecture also brings clients closer to the resources they're interacting with and opens the door for further DNS-based functionality. Many examples of such functionality will surely emerge as DNS encryption and other confidentiality protection techniques are deployed, both in a multi-resolver architecture and elsewhere.

The standards community has built a solid foundation for DNS over nearly four decades of the protocol's history. Protecting the confidentiality of DNS exchanges, through DNS encryption protocols and alternative techniques, is the most recent focus of standards efforts. These efforts balance among multiple approaches to information protection and open the door for new DNS-based functionality and standards. It will be exciting to see what comes next.

ACKNOWLEDGEMENTS

This work would not have been possible without the support of many colleagues at Verisign who have contributed to the preparation of this article. Special thanks to Scott Hollenbeck for insights into IETF standards activities; to Christine Lentz, who chairs the company's content review board, and to the members of that board; to Becca McCary for coaching on the writing process; and to Danny McPherson for guidance in developing the framework presented here. Thanks also to Guest Editor Chris Mitchell for encouragement to submit the paper and to the anonymous reviewers for their constructive comments.

REFERENCES

[1] Verisign, *The Verisign Domain Name Industry Brief, Q1 2021*, https://www.verisign.com/en_US/domain-names/dnib/index.xhtml, accessed Aug. 25, 2021.

[2] IANA, Root Zone Database. https://www.iana.org/domains/root/db, accessed Aug. 25, 2021.

[3] S. Bortzmeyer, "DNS Query Name Minimisation to Improve Privacy," RFC 7816, DOI 10.17487/RFC7816, Mar. 2016, https://www.rfc-editor.org/info/rfc7816.

[4] W.B. de Vries *et al.*, "A First Look at QNAME Minimization in the Domain Name System," in *Passive and Active Measurement, PAM 2019, Lecture Notes in Computer Science*, vol. 11419, Springer, https://doi.org/10.1007/978-3-030-15986-3_10.

[5] T.V. Doan, I. Tsareva, and V. Bajpai, "Measuring DNS over TLS from the Edge: Adoption, Reliability, and Response Times," in *Passive and Active Measurement, PAM 2021, Lecture Notes in Computer Science*, vol. 12671, Springer, https://doi.org/10.1007/978-3-030-72582-2_12.

[6] Z. Hu *et al.*, "Specification for DNS over Transport Layer Security (TLS) ," RFC 7858, DOI 10.17487/RFC7858, May 2016, https://www.rfc-editor.org/info/rfc7858.

[7] P. Hoffman and P. McManus, "DNS Queries over HTTPS (DoH)," RFC 8484, DOI 10.17487/RFC8484, Oct. 2018, https://www.rfc-editor.org/info/rfc8484.

[8] S. Bortzmeyer and S. Huque, "NXDOMAIN: There Really Is Nothing Underneath," RFC 8020, DOI 10.17487/RFC8020, Nov. 2016, https://www.rfc-editor.org/info/rfc8020.

[9] K. Fujiwara, A. Kato, and W. Kumari, "Aggressive Use of DNSSEC-Validated Cache," RFC 8198, DOI 10.17487/RFC8198, July 2017, https://www.rfc-editor.org/info/rfc8198.

[10] W. Kumari and P. Hoffman, "Running a Root Server Local to a Resolver," RFC 8806, DOI 10.17487/RFC8806, June 2020, https://www.rfc-editor.org/info/rfc8806.

[11] D. Wessels *et al.*, "Message Digest for DNS Zones," RFC 8976, DOI 10.17487/RFC8976, Feb. 2021, https://www.rfc-editor.org/info/rfc8976.

[12] E. Lewis and A. Hoenes, Ed., "DNS Zone Transfer Protocol (AXFR)," RFC 5936, DOI 10.17487/RFC5936, June 2010, https://www.rfc-editor.org/info/rfc5936.

[13] Root Server Operators, "Statement on DNS Encryption," Mar. 2021, https://root-servers.org/media/news/Statement_on_DNS_Encryption.pdf, accessed Aug. 25, 2021.

[14] C. Contavalli *et al.*, "Client Subnet in DNS Queries," RFC 7871, DOI 10.17487/RFC7871, May 2016, https://www.rfc-editor.org/info/rfc7871.

[15] National Security Agency, "Adopting Encrypted DNS in Enterprise Environments," Information sheet PP-21-0016, version 1.0, Jan. 2021. https://media.defense.gov/2021/Jan/14/2002564889/-1/-1/0/CSI_ADOPTING_ENCRYPTED_DNS_U_OO_102904_21.PDF.

[16] B. Kaliski. "Authenticated Resolution and Adaptive Resolution: Security and Navigational Enhancements to the Domain Name System," Verisign blog, Nov. 19, 2020, https://blog.verisign.com/security/authenticated-resolution-and-adaptive-resolution-security-and-navigational-enhancements-to-the-domain-name-system/, accessed Aug. 25, 2021.

BIOGRAPHY

BURTON S. KALISKI JR. [SM] (bkaliski@verisign.com) is senior vice president and chief technology officer of Verisign. He leads Verisign's long-term research program and is responsible for the company's industry standards engagements, university collaborations and technical community programs. He previously served as the founding director of the EMC Innovation Network, as vice president of research at RSA Security, and as the founding scientist of RSA Laboratories, where his contributions included the development of the Public-Key Cryptography Standards (PKCS). He received a doctorate, master's degree and bachelor's degree in computer science from the Massachusetts Institute of Technology.

PUBLICATION HISTORY

Some of the material in this article has appeared in preliminary form in one of the author's blog posts [17] and is incorporated here with permission. Figure 1 appeared in a blog post by the author's colleague [18] and is incorporated with permission. (The caption is new.)

[17] B. Kaliski. "A Balanced DNS Information Protection Strategy: Minimize at Root and TLD, Encrypt When Needed Elsewhere," Verisign blog, Dec. 7, 2020, https://blog.verisign.com/security/a-balanced-dns-information-protection-strategy-minimize-at-root-and-tld-encrypt-when-needed-elsewhere/, accessed Aug. 25, 2021.

[18] M. Thomas, "Maximizing Qname Minimization: A New Chapter in DNS Protocol Evolution," Verisign blog, Sept. 16, 2020. https://blog.verisign.com/security/maximizing-qname-minimization-a-new-chapter-in-dns-protocol-evolution/, accessed Aug. 25, 2021.