

Network Working Group
Internet-Draft
Intended status: Informational
Expires: March 13, 2020

V. Bertola
Open-Xchange
September 10, 2019

Recommendations for DNS Privacy Client Applications
draft-bertola-bcp-doh-clients-01

Abstract

This document presents operational, policy and security considerations for the authors and publishers of client applications that choose to implement DNS resolution through any of the protocols that provide private, encrypted connections between the application itself and the DNS resolver. As these protocols, depending on implementation choices and deployment models, may impact the Internet significantly at the architectural, legal and policy levels, the document records the current consensus on how these protocols should be used by applications, especially user-facing applications meant for mass usage by non-technical consumers.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on March 13, 2020.

Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents

carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
2. Requirements Notation and Conventions	3
3. Architectures for Name Resolution Services	3
4. Issues and Recommendations	5
4.1. Trust Model and User Choice	5
4.2. Consolidation	7
4.3. Competition and Network Neutrality	9
4.4. Namespace Fragmentation	10
4.5. Privacy	11
4.6. Content Access Control	12
4.7. Security and Network Management	13
4.8. Jurisdiction	15
4.9. Disaster Recovery	17
4.10. User Support	17
5. Security Considerations	18
6. Privacy Considerations	18
7. Human Rights Considerations	18
8. IANA Considerations	18
9. Acknowledgements	18
10. References	19
10.1. Normative References	19
10.2. Informative References	19
Author's Address	20

1. Introduction

As a reaction to growing concerns about widespread "pervasive monitoring" activities, the IETF declared these practices to be an attack [RFC7258] and started work to promote the encryption of the transport of information across the Internet.

The Domain Name System [RFC1034] is a fundamental element of the Internet, as almost any online activity starts with one or more DNS queries, which can be used to track the services and the content that the user is accessing, and even to redirect or disallow such access; DNS traffic deriving from the activity of human beings constitutes sensitive and valuable personal information. Section 2.4.1 of [I-D.bortzmeyer-dprive-rfc7626-bis] describes the risks created by the unencrypted transmission of such information.

To mitigate these risks, two new standards have been developed to encrypt the transport of DNS queries and replies between the user's machine and the recursive resolver: DNS-over-TLS [RFC7858] and DNS-over-HTTPS [RFC8484]. The adoption of these protocols is still limited, but early deployments, especially of the latter, have raised a number of issues that pertain to consolidation and centralization, other privacy risks, various cases of content control, network security and management, applicable jurisdiction and more.

These issues, if not addressed, could outweigh the benefits deriving from the encryption of DNS transport. Some of them can be addressed at the server side of the connection; they are the subject of [I-D.ietf-dprive-bcp-op]. However, some of these issues derive from the behaviour of client applications that implement the protocols and use them to query the DNS as necessary for their activities.

This document presents the best practices that address these issues at the client side of the connection and that, as far as possible, could allow an uncontroversial and positive deployment of encrypted DNS transport protocols.

2. Requirements Notation and Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

Throughout this document, values are quoted to indicate that they are to be taken literally. When using these values in protocol messages, the quotes MUST NOT be used as part of the value.

3. Architectures for Name Resolution Services

It is possible for a device to perform full DNS name resolution on its own, without relying on any recursive DNS resolver - indeed, this was often the case in the early usage of the DNS.

However, since when the Internet became a mass affair, the DNS name resolution service in consumer Internet access services has been generally provided by a recursive DNS resolver on the local network ("local resolver"), supplied by the same ISP that is providing the user with Internet access, often through automated configuration mechanisms such as DHCP [RFC2131].

More recently, public DNS resolvers ("remote resolvers"), provided on an Internet scale by a few big operators, have been gaining ground; users have been able to set them as the resolver for all their

applications at once, by changing a configuration item in their devices.

Thus, the current architecture for mainstream DNS resolution services relies on the following assumptions:

1. All the applications on the same device use the same resolver, through a library provided by the operating system;
2. By default, the device will automatically discover and use the resolver provided by the local network;
3. The user is in charge of deciding which resolver will be used, either by silently accepting the default, or by setting a specific resolver in the device configuration.

Early deployments of DNS-over-TLS have generally followed the same model; in the first mobile operating system that has implemented support for the protocol, the system will try to discover whether the resolver configured in the operating system supports DNS-over-TLS connections, and if so, it will automatically upgrade the connection to the new protocol while continuing to use the same resolver; otherwise, it will keep using the unencrypted connection.

Early deployments of DNS-over-HTTPS have followed a different model. The first major Web browser releasing support for DNS-over-HTTPS has announced the intention to follow a service architecture based on different assumptions:

1. Each application will use its own resolver, bypassing the library provided by the operating system;
2. No automatic discovery of resolvers on the local network is performed;
3. The application is in charge of determining which resolver will be used, which could be different than the one configured in the operating system, and can direct this choice by selecting and providing one as the default for all users globally, or even by limiting the user's choice to a list of resolvers vetted by the application maker.

In the rest of the document, we will refer to this new architecture as "application-level name resolution", and to the traditional one as "network-level name resolution". More specifically, we will refer to point 3 of the above list as "application-level resolver selection".

While the document will also address the issues that derive simply from the switch to an encrypted connection, most of the issues that have been noticed during the early deployment efforts for DNS-over-HTTPS do not derive from the protocol in itself, but rather from the switch to application-level name resolution under the assumptions above, and most frequently from the adoption of application-level resolver selection.

Other possible architectures have also been identified: for example, applications could use different resolvers for each of the different servers that they have to connect to, or the servers could push DNS records to the client even before they are actually queried. As these models are still speculative, the issues that they would generate are not currently addressed by this document.

4. Issues and Recommendations

This section classifies the issues that are created by the deployment of encrypted DNS protocols and/or by the change of resolution architectures on the client side, and presents recommendations to address them.

4.1. Trust Model and User Choice

With network-level name resolution, users are in charge of the choice of the resolver used by all their applications. Advanced and educated users make full use of that opportunity, and choose to send their DNS queries to an operator they trust, either by configuring a specific resolver in the operating system or on the router that manages their home network, or as part of a broader traffic encryption and redirection service (e.g. VPNs). Other users will just rely on the local network's server; for home users, this will be provided by their Internet access provider, which they also picked, thus giving them at least some degree of trust. A different, less trusted relationship exists when users that did not configure a specific server connect to a local network which is not their own, for example in Internet cafes or hotels; in that case, they may be led to use a resolver which they do not trust.

Also, in the absence of encryption, the local network operator could still be able to track or even alter the DNS queries and replies that the user has directed to a non-local resolver. Unless this has been agreed with the user or is mandated by applicable legislation, this would be a breach of the user's trust and is a good reason to promote the adoption of encrypted DNS protocols.

With application-level name resolution, and especially with application-level resolver selection, users lose at least part of

their control. Some applications may actually not give the user any choice, and just use their own resolver all the times; some others may provide configuration options, but still direct the queries to their own resolver as a default, requiring specific action for users to keep their DNS traffic going to the resolver that they already configured in the operating system.

Such a change of default behaviour would break the expectations of many users; unless appropriate communication is given and consent is acquired, both the users that explicitly configured a resolver in the operating system, and the users that want their device to use the local network's resolver, will be surprised by the application sending DNS queries to another server instead. On the other hand, for less technical users that trust the application maker to make a better choice of resolvers on their behalf, the new behaviour could be in line with their expectations.

However, also given the high sensitivity of the personal information embedded in DNS queries, it is important to ensure that it is ultimately the user, rather than any third party, that determines where their DNS queries should go, and explicitly consents to any choice of resolver that cannot be expected, or to delegating the choice to another party.

This leads to the following recommendations:

1. Users **MUST** have the final word on which resolver is used by each application.
2. Applications **MUST NOT** use a different resolver operator than the one that would be used by the operating system, unless the user has actively told the application to use a different resolver operator and has given explicit and informed consent to the change.
3. Applications **MUST** provide users with a configuration mechanism that allows them to tell the application to use any resolver the user wants.
4. Applications **SHOULD** provide users with adequate information on the location, ownership, data management policies and operational practices of each resolver, at least for the resolvers that they provide as hard-coded configuration options.
5. Applications **SHOULD** support, if available, easy ways for the user to direct all applications on the device to use a specific resolver and a specific protocol, without the need to configure them one by one.

For recommendation #2, it may happen that the resolver configured in the operating system does not support encrypted DNS protocols. In this case, the application SHOULD first of all try to determine whether the operator of the non-encrypted resolver that would be used by the operating system also provides any encrypted DNS resolver, through standardized discovery mechanisms such as [I-D.ietf-doh-resolver-associated-doh]; in that case, it SHOULD use that operator's encrypted resolver instead of the unencrypted one.

For recommendation #4, the guidelines in [I-D.ietf-dprive-bcp-op] SHOULD be followed; applications could explore ways to make them more understandable to average Internet users, for example through the use of standardized visual aids. At the same time, due to the complexity and changing nature of this information, applications could simply provide links to where the operator makes it available; a standardized and automated way for resolvers to make this information available to applications would be desirable.

For recommendation #5, in most operating systems and devices it is yet to be discussed how to achieve the objective of letting the user set the preferred resolver and protocol in all the applications at once; the principle, however, deserves to be stated as a recommended direction for future development.

4.2. Consolidation

Currently, with network-level name resolution, DNS resolution activities are globally spread across a huge number of resolvers, possibly in the range of the hundreds of thousands or even millions; while some consolidation is happening, mostly into some big remote resolver platforms, still it takes the thousand biggest resolvers to aggregate 90% of global DNS traffic. Users that keep the default of using the local network's resolver see their personal DNS queries spread across multiple servers as they move; users that configure a specific public resolver have their queries concentrated on a single resolver system, but they can pick one that they trust.

However, in many cases, the global market for user applications is much more consolidated than the global market for Internet access and for DNS resolution services. More specifically, estimates for the global browser market currently show one maker having around 60% of the market, and the first four together having over 90% of it. While these market shares may change in the future, the presence of one or a few dominant browsers has almost always been the case since the advent of mass usage of the World Wide Web.

As application-level name resolution turns the application maker into a potential gatekeeper in the choice of the resolver, there is a

concern that each application maker could lead its users to the adoption of one specific resolver operator, or limit their choice to a few options that the maker has picked for its own reasons.

While it is understandable that an application maker may want to help its users make a good choice of resolver by using its technical expertise to evaluate and select a narrow set of recommended resolvers, this practice would open opportunities for misuse, as applications could recommend resolvers not because of a fair evaluation, but because of an existing business partnership in the mutual economic interest, or even because the resolver is run directly by the application maker.

In the end, this could lead to a dramatic consolidation of global name resolution operators, with a few server systems managing the broad majority of global resolutions. This, in turn, would have several negative consequences, which will be discussed in the following sections of the document.

However, consolidation is an architectural problem in itself; the Internet was designed to be as decentralized as possible, to increase its resilience and ultimately the freedom of its users. The concern over the centralizing effect of encrypted DNS protocols has been recorded for example in [I-D.arkko-iab-internet-consolidation], section 2.5, and has to be addressed by preventing the use of a strong position in the market for a specific type of client application, especially a ubiquitous one such as browsers, to centralize DNS resolutions and establish a strong position in DNS name resolution services.

This would already be addressed by the recommendations in section 4.1, but it also leads to the following additional recommendations:

1. Applications MUST NOT prioritize any specific resolver over others unless told so by the user, or design their user interaction to lead users towards choosing a specific resolver or one of a few specific resolvers.
2. If possible, and if desired by the user, applications SHOULD provide ways to spread their DNS resolution traffic across the highest possible number of recursive resolvers. [NOTE: this recommendation is actually a placeholder, as there are drawbacks to this idea and other, better methods could be devised - this is TBD]
3. If applications would like to ensure that their users can pick a resolver that has been vetted under a set of criteria, the definition, verification and enforcement of these criteria SHOULD

be deferred to an independent multi-stakeholder organization, making these criteria public and objective and giving any resolver operator from any part of the Internet a fair chance to be admitted to the list of vetted services; in any case, users MUST still be free to adopt servers outside of the vetted list if they so desire.

4.3. Competition and Network Neutrality

The possible inception of a stricter relationship between application makers and resolver operators also creates issues related to competition, which can in turn have effects on the architecture and performance of the network and on its capacity for permissionless innovation, and compromise the neutrality of network transport, as DNS resolution is a fundamental step in establishing any network connection.

In addition to the consolidation effect described in the previous section, any relationship between the application maker and a resolver operator that could lead the application to direct users to a specific resolver could be used to reduce the opportunities available to other resolver operators, including innovative ones, and even to other application makers, for example by virtue of exclusive service agreements. This can be a likely case when the application maker and the resolver operator are the same entity, but it could happen also when the application maker and the resolver operator are two different entities that have established a business partnership in the mutual interest.

Another case of anti-competitive effect could happen when the resolver operator also operates another service that depends for its effectiveness on the results of the DNS resolution process - for example, content delivery networks. Resolver operators that also operate CDNs could provide slower or less effective DNS resolution for access to content distributed by their competitors.

While this matter will also be under the purview of national and international market regulations, there are some recommendations - in addition to those in the previous section, that also address this issue - that can be made in terms of best practices, to neutralize as far as possible any direct relationship between the application maker and the resolver operator:

[to be discussed]

4.4. Namespace Fragmentation

If each application could pick a different resolver, there is a chance that different applications would receive different replies to the same DNS queries, leading to confusing user experiences, potential attack surfaces and network management and debugging problems, and in the end to the fragmentation of the Internet into non fully interoperable chunks.

The only way to prevent any occurrence of this case would be to require all applications on the same device to use the same resolver, as in the current network-level resolution model; however, such requirement would be considered too restrictive. Recommendations #2 and #3 in section 4.1 are however meant to make this case a special exception, rather than the norm, and thus mitigate this problem.

However, when allowing different applications to use different resolvers, there is a situation that would significantly endanger the stability of the Internet. Currently, as the application and the resolver are generally provided by two independent entities, and as the application cannot know in advance which resolver will be used, applications cannot rely on predetermined non-standard behaviour by a specific resolver. With application-level resolver selection, applications that enjoy a significant market share could direct users to resolvers that employ alternate DNS namespaces that they control, or start to create and remove top level domains outside of the collectively agreed policy frameworks.

The global uniqueness of the DNS namespace and its collective policy-making procedures should be preserved, and this leads to the following additional recommendation:

1. Applications SHOULD NOT adopt alternate DNS namespaces or promote the use of resolvers that do not rely on the global DNS root server system.
2. Applications SHOULD NOT deviate from the DNS namespace management policies, technical standards and operational practices that are defined in the relevant Internet governance venues.

Regarding recommendation #1, there may be specific use cases in which a user may want to use an alternate namespace and root server set, and applications, if they want, should be free to support them; however, these cases must be exceptions and not for mainstream usage.

4.5. Privacy

Protecting the user's privacy is the primary aim of transitioning the DNS to encrypted communications. However, risks to privacy deriving from DNS communications are not limited to the eavesdropping of unencrypted DNS communications; [I-D.bortzmeyer-dprive-rfc7626-bis] lists, in section 2.4.2, several privacy risks that still exist even when DNS communications are encrypted; and, in section 2.5.1, it describes the risks that derive from the behaviour of the resolver in use, independently from whether the connection is encrypted or not.

To address the former category of privacy risks, some have suggested that, through the use of DNS-over-HTTPS, the user's requests could be hidden within unrelated HTTPS traffic, locating DNS-over-HTTPS servers at the same IP address and port of widely used web servers. However, this method creates security and legal issues that are documented in the following sections of this document, and so it is not recommended unless the privacy gain must prevail upon any other consideration, for example because of the extreme sensitivity of the online activity or because of a hostile environment that could lead to significant personal risks.

In all other cases, users that feel the need for further obfuscation of their encrypted DNS communication should rather be directed to spreading their communications across a great number of encrypted resolvers, as per recommendation #2 of section 4.2, making it harder to track the entirety of their DNS exchanges. Moreover, if privacy is so important, users should consider that they also need to encrypt the rest of the communications, and not just the resolution of domain names; thus, using a VPN for all their traffic might just be easier and better.

To address the latter category of privacy risks, the first and foremost mitigation measure is to allow each user to pick a resolver that is managed by an organization that they trust, under appropriate regulatory conditions, privacy policies and operational practices. While "privacy-friendly" applications might (and, indeed, should) help users in making this choice, there is not a unique, globally applicable agreement on what constitutes a "privacy-friendly" resolver, and it is hard to ensure that all application makers will always make disinterested choices in the pure interest of the user; so it must be the user, in the end, to decide who to entrust with their personal information. The recommendations in sections 4.1 and 4.2 thus also contribute to mitigate this type of risks.

In addition, specific technical recommendations can be made to prevent applications from adopting practices that would make it easier for the resolver operator to track and fingerprint the user,

mirroring the ones that have been developed in [I-D.ietf-dprive-bcp-op] section 5:

1. Applications MUST authenticate the resolver they connect to, if they have securely discovered the information required to do so.
2. In the case of DNS-over-HTTPS, applications SHOULD NOT attach or receive HTTP cookies on the connections used for the DNS message exchange, unless there is a specific use case for cookies that has been explicitly requested by the user.
3. [to be continued]

For recommendation #1, and for DNS-over-TLS, a discussion of resolver authentication methods and possibilities can be found in [RFC8310].

For recommendation #2, additional considerations on privacy when using HTTPS as a transport mechanism for other protocols can be found in section 6.1 of [I-D.ietf-httpbis-bcp56bis].

4.6. Content Access Control

DNS name resolution services are commonly chosen as a platform to control user access to content; while there are other mechanisms in use, checking and blocking destinations at the DNS level is an effective and relatively inexpensive one. As a consequence, the choice of the resolver also determines whether the user will be able to access certain destinations or not, depending on the policies applied by the individual resolver.

Sometimes, on unencrypted DNS connections, content control policies will also be applied to DNS connections directed to other resolvers having a different policy than the local network's one, though this is made impossible by the switch to encrypted DNS connections.

The motivations, the procedures, and the types of content subject to blocking are very variable. Some of the filtering activities are meant to increase the security and health of the network; they can be aimed at non-human clients, such as bots, trying to prevent them from connecting to their command and control servers; or they can try to prevent unaware users from connecting to phishing and malware websites.

Other filtering activities are meant to make some content inaccessible because it violates the law in the user's and/or the resolver's jurisdiction, or because of court rulings that mandate the block. In authoritarian countries, content may be blocked to prevent criticism of the ruling entity, while in democratic countries it can

be blocked to defend the safety and rights of some parties and prevent violence and attacks on democracy. Destinations may also be made inaccessible, independently from their content, for lack of compliance with any applicable regulation, such as requirements for licenses or the payment of taxes in the user's country.

The opinions on the appropriateness of these practices are highly influenced by local culture, history and socio-political environments, as well as by each individual's set of values, and it is thus impossible to make blanket recommendations on whether and when they should be forbidden, allowed, or actively supported; the only possible recommendation is to follow the applicable regulations.

In some cases, however, users actually demand DNS-based content control services to shape their own Internet access: for example, families that want to make sure that their children using the Internet from home cannot access inappropriate websites; businesses that want to restrict the way their employees can use the Internet in the office during working hours.

This leads to the following recommendations:

1. Applications **MUST NOT** prevent users from using any DNS-based content control service that they freely choose to adopt.
2. Applications **SHOULD** comply with any legislation and legal ruling on content control that applies to them in the jurisdiction where they are being used.

Further considerations on the relationship between applications and applicable legislation will be made in section 4.7.

4.7. Security and Network Management

Network management and network security practices often rely on checks and policies implemented at the DNS level, through the monitoring and mangling of the DNS queries that the users of the local network send to the local resolver. For example, these practices include checking for any unusual patterns in DNS traffic to detect devices that have been infected with malware; blocking queries for known botnet command-and-control servers to make it impossible for bots to communicate with them and take orders; implementing a content control list of web destinations that the network administrator considers dangerous; associating certain names to different IP addresses, some of which may be private, depending on the client and on its topological location; creating names in special use or non-standard top level domains and making them resolvable only from the inside of the local network.

Especially the use of local names and "split horizon" configurations is a widely used security practice in corporate network environments; using a resolver located outside of the network would break this mechanism and at the same time leak information that would be very valuable to an attacker.

These practices rely on forcing all the users of the local network to use the local resolver, rather than a remote public resolver; this requirement is generally enforced through one or more of the following practices:

- a. Making sure that all devices that are connected to the local network are configured to use the local resolver, disallowing the users from modifying this configuration entry;
- b. Blocking access to remote resolvers at the edge of the local network, for example through firewalls and blocks by destination IP address and/or port;
- c. Examining, and if necessary amending, all DNS queries and replies in transit towards remote resolvers.

The switch to encrypted DNS connections makes practice c) generally impossible; the switch to application-level resolver selection also makes a) impossible, unless the network administrator can prevent the installation of any application on all the devices connected to the network, which is not easily feasible in most cases. Finally, the implementation of "mixed mode" DNS-over-HTTPS, obfuscating the traffic within ordinary HTTPS connections to widely used web hosts, would make also practice b) impossible.

So, the deployment of encrypted DNS over dedicated connections disables c), but still leaves a) and b) as options to network administrators; however, the deployment of DNS-over-HTTPS in mixed mode disables all three practices and leaves the network administrators powerless; additionally, it even provides an undetectable data exfiltration vector for malicious applications that are trying to steal confidential information from the local network and forward it to the outside.

While disabling control by the local network administrator might actually be a positive intent in very specific cases, disabling all these security practices at once is dangerous and inappropriate in most cases. It is especially problematic on private networks that host sensitive or commercially valuable information, as they need to be able to provide some degree of connectivity to the Internet while scrutinizing and limiting its usage to guarantee security.

Noting that even [RFC7258], in section 2, stresses that "Making networks unmanageable to mitigate pervasive monitoring is not an acceptable outcome", and calls for "an appropriate balance" between encryption and network management needs, the following recommendations, in addition to those in section 4.1., represent such balance:

1. Applications SHOULD NOT adopt the practice of hiding their encrypted DNS traffic in ways that prevent the local network administrator from isolating it, monitoring it (without breaking the encryption) and, if desired, blocking it; exceptions to this principle MUST be motivated by a clear and compelling use case which cannot be addressed otherwise, and their use MUST be limited to such use case. [Note: of course I am sure that many people will want to discuss this - the idea is to not stop this from happening when it is really useful, but also restrict its usage to when it is really useful, leaving network admins with the possibility to block external encrypted resolvers if they want, without starting a technical "arms race".]
2. [to be continued]

Further reasons supporting recommendation #1 can be found in sections 4.5 and 4.8.

4.8. Jurisdiction

The current prevailing practice of using local resolvers, located topologically near to the user, generally ensures that the resolver and the user fall under the same jurisdiction. This allows the country managing such jurisdiction to apply rules and policies to the user's Internet access by acting upon the resolver, recommending or mandating actions to the ISP that runs the resolver.

The use of remote resolvers, in most cases located in a different country and under a different jurisdiction than the user, has already provided a way for users to bypass their local laws. Many countries have tolerated this loss of sovereignty because it only affected a minority of users, possibly smarter technical users that could have anyway found other ways to bypass national rules applied at the resolver level, such as running their own recursive resolver. Other countries have instead engaged in enforcing their Internet access rules at other levels, such as by requiring firewalls at each connection between any national network and the broader Internet and filtering out destinations by IP address, or requiring the ISPs to apply similar blocks on each user's Internet connectivity.

In [RFC7754], the IETF has recommended the use of filtering at the endpoints, rather than inside the network, to address issues that require access control to Internet destinations; but filtering at the edges is much harder to set up and to enforce for countries, and the same RFC agrees that "a hybrid approach that combines endpoint-based filtering with network filtering may prove least damaging". In this regard, making resolver-level law-mandated filtering policies impossible is anyway likely to push more countries to mandate heavier, more damaging filtering practices at the IP address level.

From the user's viewpoint, a change in jurisdiction may be beneficial or damaging, depending on which issues are most important for the user and on the applicable legislation in the user's and, if different, in the resolver's country. Users located in jurisdictions that restrict their rights may actively seek to use a resolver in a different jurisdiction to bypass the restrictions. Users that enjoy a high degree of rights in their country, such as extended protections for privacy and network neutrality, may reject the use of a resolver in another jurisdiction not to lose such protections.

It is out of scope for the IETF to decide whether the policies and laws of any country should be supported or opposed. By leaving as much control as possible to the user, as recommended in section 4.1, each user is allowed to decide whether to seek the use of a resolver in a different jurisdiction or stay within their own, bearing the responsibility for any legal consequence that might derive from that decision; and it is important that such a decision is not taken lightly and especially is not taken by the application without telling the user, as the user could otherwise unwillingly end up in legal troubles.

At the same time, while individual users and individual application makers may have different views and follow other practices, it is inappropriate for the IETF as a whole to promote the deployment of technologies in a way that is explicitly designed to undermine any country's sovereignty and jurisdiction. This is another reason to support recommendation #1 in section 4.7 and recommendation #2 in section 4.6. More generally, the following recommendations can be devised:

1. Applications SHOULD clearly inform the user whenever the resolver that it is going to be employed lies under a jurisdiction different than the one of the user.
2. Applications SHOULD NOT adopt a resolver located under a jurisdiction different than the user's one, unless the user has explicitly consented to such change of jurisdiction, and unless

the user's jurisdiction allows the use of resolvers located in a different country.

4.9. Disaster Recovery

Remote resolvers rely on global Internet connectivity to be able to serve users from every part of the world. However, there may be cases in which long distance Internet connectivity is so poor to make the use of remote resolvers ineffective, or has been greatly reduced or even completely severed by the effects of natural disasters, upstream legal and contractual issues or any other special situation.

In these cases, it is important that applications can continue working if connectivity to a local resolver can be established, as the resolver, even with global connectivity issues, can still provide access to much needed local resources.

This leads to the following recommendation:

1. Applications, when using a remote resolver, SHOULD monitor the availability of sufficient connectivity to it, and SHOULD prompt the user to switch temporarily to a local resolver, if available, when such connectivity is not available.

4.10. User Support

Functioning DNS resolution is a requirement to make Internet connectivity work, and, when it does not, most users have a hard time discerning the specific technical factor that prevents it from working. As they generally acquire their Internet connectivity from a local ISP, they will thus contact the ISP's support desk whenever the connectivity does not work, regardless of the actual cause. However, if the application is using a remote resolver, the ISP's support desk will not be able to know whether such resolver is experiencing operational issues or applying policies that make the user's desired action fail.

It is thus important that application makers, remote resolver operators and ISPs cooperate to allow users to get support on their Internet access problems. For what regards applications, this leads to the following recommendations:

1. Applications SHOULD make it easy for users to determine whether any failure that they experience in the use of the application can be attributed to a failure in DNS resolution.

2. Applications SHOULD cooperate with remote resolver operators to direct users that experience problems due to resolver failures to the support service of the resolver operator.
3. Applications SHOULD provide easy ways for their users to retrieve the information on the resolver currently in use, so that they can pass on this information to whoever is helping them to restore their connectivity.

5. Security Considerations

The use of encrypted DNS protocols is beneficial to security as it prevents unauthorized third parties from altering the DNS queries and replies, but it also creates new security risks by disrupting a number of existing and commonly used practices for network security, and by providing, under certain conditions, a mechanism for data exfiltration from within a network through the submission of appropriately designed DNS queries.

More detailed security considerations can be found in section 4.7 of this document.

6. Privacy Considerations

The use of encrypted DNS protocols in itself is beneficial to privacy as it prevents eavesdropping of the connection. However, the issues created by the deployment model can lead to an overall loss of privacy, for example because the user is led to adopt a resolver operator that offers less privacy protection than the one they are currently using, or that is located under a jurisdiction that offers less privacy protection.

Issues specifically related to privacy, and recommendations to address them, are discussed in section 4.5 of this document.

7. Human Rights Considerations

[to be written]

8. IANA Considerations

This document has no actions for IANA.

9. Acknowledgements

[to be written]

10. References

10.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.

10.2. Informative References

- [I-D.arkko-iab-internet-consolidation]
Arkko, J., Trammell, B., Nottingham, M., Huitema, C., Thomson, M., Tantsura, J., and N. Oever, "Considerations on Internet Consolidation and the Internet Architecture", draft-arkko-iab-internet-consolidation-02 (work in progress), July 2019.
- [I-D.bortzmeyer-dprive-rfc7626-bis]
Bortzmeyer, S. and S. Dickinson, "DNS Privacy Considerations", draft-bortzmeyer-dprive-rfc7626-bis-02 (work in progress), January 2019.
- [I-D.ietf-doh-resolver-associated-doh]
Hoffman, P., "Associating a DoH Server with a Resolver", draft-ietf-doh-resolver-associated-doh-03 (work in progress), March 2019.
- [I-D.ietf-dprive-bcp-op]
Dickinson, S., Overeinder, B., Rijswijk-Deij, R., and A. Mankin, "Recommendations for DNS Privacy Service Operators", draft-ietf-dprive-bcp-op-03 (work in progress), July 2019.
- [I-D.ietf-httpbis-bcp56bis]
Nottingham, M., "Building Protocols with HTTP", draft-ietf-httpbis-bcp56bis-08 (work in progress), November 2018.
- [RFC1034] Mockapetris, P., "Domain names - concepts and facilities", STD 13, RFC 1034, DOI 10.17487/RFC1034, November 1987, <<https://www.rfc-editor.org/info/rfc1034>>.
- [RFC2131] Droms, R., "Dynamic Host Configuration Protocol", RFC 2131, DOI 10.17487/RFC2131, March 1997, <<https://www.rfc-editor.org/info/rfc2131>>.

- [RFC7258] Farrell, S. and H. Tschofenig, "Pervasive Monitoring Is an Attack", BCP 188, RFC 7258, DOI 10.17487/RFC7258, May 2014, <<https://www.rfc-editor.org/info/rfc7258>>.
- [RFC7754] Barnes, R., Cooper, A., Kolkman, O., Thaler, D., and E. Nordmark, "Technical Considerations for Internet Service Blocking and Filtering", RFC 7754, DOI 10.17487/RFC7754, March 2016, <<https://www.rfc-editor.org/info/rfc7754>>.
- [RFC7858] Hu, Z., Zhu, L., Heidemann, J., Mankin, A., Wessels, D., and P. Hoffman, "Specification for DNS over Transport Layer Security (TLS)", RFC 7858, DOI 10.17487/RFC7858, May 2016, <<https://www.rfc-editor.org/info/rfc7858>>.
- [RFC8310] Dickinson, S., Gillmor, D., and T. Reddy, "Usage Profiles for DNS over TLS and DNS over DTLS", RFC 8310, DOI 10.17487/RFC8310, March 2018, <<https://www.rfc-editor.org/info/rfc8310>>.
- [RFC8484] Hoffman, P. and P. McManus, "DNS Queries over HTTPS (DoH)", RFC 8484, DOI 10.17487/RFC8484, October 2018, <<https://www.rfc-editor.org/info/rfc8484>>.

Author's Address

Vittorio Bertola
Open-Xchange
Via Treviso 12
Torino 10144
Italy

Email: vittorio.bertola@open-xchange.com
URI: <https://www.open-xchange.com>

dprive
Internet-Draft
Obsoletes: 7626 (if approved)
Intended status: Informational
Expires: July 19, 2019

S. Bortzmeyer
AFNIC
S. Dickinson
Sinodun IT
January 15, 2019

DNS Privacy Considerations
draft-bortzmeyer-dprive-rfc7626-bis-02

Abstract

This document describes the privacy issues associated with the use of the DNS by Internet users. It is intended to be an analysis of the present situation and does not prescribe solutions. This document obsoletes RFC 7626.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on July 19, 2019.

Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
2. Risks	4
2.1. The Alleged Public Nature of DNS Data	5
2.2. Data in the DNS Request	5
2.2.1. Data in the DNS payload	7
2.3. Cache Snooping	7
2.4. On the Wire	7
2.4.1. Unencrypted Transports	7
2.4.2. Encrypted Transports	9
2.5. In the Servers	10
2.5.1. In the Recursive Resolvers	10
2.5.2. In the Authoritative Name Servers	12
2.5.3. Rogue Servers	13
2.5.4. Authentication of servers	13
2.5.5. Blocking of services	14
2.6. Re-identification and Other Inferences	14
2.7. More Information	15
3. Actual "Attacks"	15
4. Legalities	15
5. Security Considerations	16
6. Acknowledgments	16
7. Changelog	16
8. References	17
8.1. Normative References	17
8.2. Informative References	17
8.3. URIs	22
Authors' Addresses	22

1. Introduction

This document is an analysis of the DNS privacy issues, in the spirit of Section 8 of [RFC6973].

The Domain Name System is specified in [RFC1034], [RFC1035], and many later RFCs, which have never been consolidated. It is one of the most important infrastructure components of the Internet and often ignored or misunderstood by Internet users (and even by many professionals). Almost every activity on the Internet starts with a DNS query (and often several). Its use has many privacy implications and this is an attempt at a comprehensive and accurate list.

Let us begin with a simplified reminder of how the DNS works. (See also [RFC8499]) A client, the stub resolver, issues a DNS query to a server, called the recursive resolver (also called caching resolver or full resolver or recursive name server). Let's use the query "What are the AAAA records for www.example.com?" as an example. AAAA

is the QTYPE (Query Type), and `www.example.com` is the QNAME (Query Name). (The description that follows assumes a cold cache, for instance, because the server just started.) The recursive resolver will first query the root name servers. In most cases, the root name servers will send a referral. In this example, the referral will be to the `.com` name servers. The resolver repeats the query to one of the `.com` name servers. The `.com` name servers, in turn, will refer to the `example.com` name servers. The `example.com` name server will then return the answer. The root name servers, the name servers of `.com`, and the name servers of `example.com` are called authoritative name servers. It is important, when analyzing the privacy issues, to remember that the question asked to all these name servers is always the original question, not a derived question. The question sent to the root name servers is "What are the AAAA records for `www.example.com`?", not "What are the name servers of `.com`?". By repeating the full question, instead of just the relevant part of the question to the next in line, the DNS provides more information than necessary to the name server.

Because DNS relies on caching heavily, the algorithm described just above is actually a bit more complicated, and not all questions are sent to the authoritative name servers. If a few seconds later the stub resolver asks the recursive resolver, "What are the SRV records of `_xmpp-server._tcp.example.com`?", the recursive resolver will remember that it knows the name servers of `example.com` and will just query them, bypassing the root and `.com`. Because there is typically no caching in the stub resolver, the recursive resolver, unlike the authoritative servers, sees all the DNS traffic. (Applications, like web browsers, may have some form of caching that does not follow DNS rules, for instance, because it may ignore the TTL. So, the recursive resolver does not see all the name resolution activity.)

It should be noted that DNS recursive resolvers sometimes forward requests to other recursive resolvers, typically bigger machines, with a larger and more shared cache (and the query hierarchy can be even deeper, with more than two levels of recursive resolvers). From the point of view of privacy, these forwarders are like resolvers, except that they do not see all of the requests being made (due to caching in the first resolver).

Almost all this DNS traffic is currently sent in clear (unencrypted). At the time of writing there is increasing deployment of DNS-over-TLS [RFC7858] and work underway on DoH [RFC8484]. There are a few cases where there is some alternative channel encryption, for instance, in an IPsec VPN, at least between the stub resolver and the resolver.

Today, almost all DNS queries are sent over UDP [thomas-ditl-tcp]. This has practical consequences when considering encryption of the

traffic as a possible privacy technique. Some encryption solutions are only designed for TCP, not UDP.

Another important point to keep in mind when analyzing the privacy issues of DNS is the fact that DNS requests received by a server are triggered by different reasons. Let's assume an eavesdropper wants to know which web page is viewed by a user. For a typical web page, there are three sorts of DNS requests being issued:

Primary request: this is the domain name in the URL that the user typed, selected from a bookmark, or chose by clicking on an hyperlink. Presumably, this is what is of interest for the eavesdropper.

Secondary requests: these are the additional requests performed by the user agent (here, the web browser) without any direct involvement or knowledge of the user. For the Web, they are triggered by embedded content, Cascading Style Sheets (CSS), JavaScript code, embedded images, etc. In some cases, there can be dozens of domain names in different contexts on a single web page.

Tertiary requests: these are the additional requests performed by the DNS system itself. For instance, if the answer to a query is a referral to a set of name servers, and the glue records are not returned, the resolver will have to do additional requests to turn the name servers' names into IP addresses. Similarly, even if glue records are returned, a careful recursive server will do tertiary requests to verify the IP addresses of those records.

It can be noted also that, in the case of a typical web browser, more DNS requests than strictly necessary are sent, for instance, to prefetch resources that the user may query later or when autocompleting the URL in the address bar. Both are a big privacy concern since they may leak information even about non-explicit actions. For instance, just reading a local HTML page, even without selecting the hyperlinks, may trigger DNS requests.

For privacy-related terms, we will use the terminology from [RFC6973].

2. Risks

This document focuses mostly on the study of privacy risks for the end user (the one performing DNS requests). We consider the risks of pervasive surveillance [RFC7258] as well as risks coming from a more focused surveillance. Privacy risks for the holder of a zone (the risk that someone gets the data) are discussed in [RFC5936] and

[RFC5155]. Non-privacy risks (such as cache poisoning) are out of scope.

2.1. The Alleged Public Nature of DNS Data

It has long been claimed that "the data in the DNS is public". While this sentence makes sense for an Internet-wide lookup system, there are multiple facets to the data and metadata involved that deserve a more detailed look. First, access control lists and private namespaces notwithstanding, the DNS operates under the assumption that public-facing authoritative name servers will respond to "usual" DNS queries for any zone they are authoritative for without further authentication or authorization of the client (resolver). Due to the lack of search capabilities, only a given QNAME will reveal the resource records associated with that name (or that name's non-existence). In other words: one needs to know what to ask for, in order to receive a response. The zone transfer QTYPE [RFC5936] is often blocked or restricted to authenticated/authorized access to enforce this difference (and maybe for other reasons).

Another differentiation to be considered is between the DNS data itself and a particular transaction (i.e., a DNS name lookup). DNS data and the results of a DNS query are public, within the boundaries described above, and may not have any confidentiality requirements. However, the same is not true of a single transaction or a sequence of transactions; that transaction is not / should not be public. A typical example from outside the DNS world is: the web site of Alcoholics Anonymous is public; the fact that you visit it should not be.

2.2. Data in the DNS Request

The DNS request includes many fields, but two of them seem particularly relevant for the privacy issues: the QNAME and the source IP address. "source IP address" is used in a loose sense of "source IP address + maybe source port", because the port is also in the request and can be used to differentiate between several users sharing an IP address (behind a Carrier-Grade NAT (CGN), for instance [RFC6269]).

The QNAME is the full name sent by the user. It gives information about what the user does ("What are the MX records of example.net?" means he probably wants to send email to someone at example.net, which may be a domain used by only a few persons and is therefore very revealing about communication relationships). Some QNAMEs are more sensitive than others. For instance, querying the A record of a well-known web statistics domain reveals very little (everybody visits web sites that use this analytics service), but querying the A

record of `www.verybad.example` where `verybad.example` is the domain of an organization that some people find offensive or objectionable may create more problems for the user. Also, sometimes, the QNAME embeds the software one uses, which could be a privacy issue. For instance, `_ldap._tcp.Default-First-Site-Name._sites.gc._msdcs.example.org`. There are also some BitTorrent clients that query an SRV record for `_bittorrent-tracker._tcp.domain.example`.

Another important thing about the privacy of the QNAME is the future usages. Today, the lack of privacy is an obstacle to putting potentially sensitive or personally identifiable data in the DNS. At the moment, your DNS traffic might reveal that you are doing email but not with whom. If your Mail User Agent (MUA) starts looking up Pretty Good Privacy (PGP) keys in the DNS [RFC7929], then privacy becomes a lot more important. And email is just an example; there would be other really interesting uses for a more privacy- friendly DNS.

For the communication between the stub resolver and the recursive resolver, the source IP address is the address of the user's machine. Therefore, all the issues and warnings about collection of IP addresses apply here. For the communication between the recursive resolver and the authoritative name servers, the source IP address has a different meaning; it does not have the same status as the source address in an HTTP connection. It is now the IP address of the recursive resolver that, in a way, "hides" the real user. However, hiding does not always work. Sometimes EDNS(0) Client subnet [RFC7871] is used (see its privacy analysis in [denis-edns-client-subnet]). Sometimes the end user has a personal recursive resolver on her machine. In both cases, the IP address is as sensitive as it is for HTTP [sidn-entrada].

A note about IP addresses: there is currently no IETF document that describes in detail all the privacy issues around IP addressing. In the meantime, the discussion here is intended to include both IPv4 and IPv6 source addresses. For a number of reasons, their assignment and utilization characteristics are different, which may have implications for details of information leakage associated with the collection of source addresses. (For example, a specific IPv6 source address seen on the public Internet is less likely than an IPv4 address to originate behind a CGN or other NAT.) However, for both IPv4 and IPv6 addresses, it's important to note that source addresses are propagated with queries and comprise metadata about the host, user, or application that originated them.

2.2.1. Data in the DNS payload

At the time of writing there are no standardized client identifiers contained in the DNS payload itself (ECS [RFC7871] while widely used is only of Category Informational).

DNS Cookies [RFC7873] are a lightweight DNS transaction security mechanism that provides limited protection against a variety of increasingly common denial-of-service and amplification/forgery or cache poisoning attacks by off-path attackers. It is noted, however, that they are designed to just verify IP addresses (and should change once a client's IP address changes), they are not designed to actively track users (like HTTP cookies).

There are anecdotal accounts of MAC addresses [1] and even user names being inserted in non-standard EDNS(0) options for stub to resolver communications to support proprietary functionality implemented at the resolver (e.g. parental filtering).

2.3. Cache Snooping

The content of recursive resolvers' caches can reveal data about the clients using it (the privacy risks depend on the number of clients). This information can sometimes be examined by sending DNS queries with RD=0 to inspect cache content, particularly looking at the DNS TTLs [grangeia.snooping]. Since this also is a reconnaissance technique for subsequent cache poisoning attacks, some counter measures have already been developed and deployed.

2.4. On the Wire

2.4.1. Unencrypted Transports

For unencrypted transports, DNS traffic can be seen by an eavesdropper like any other traffic. (DNSSEC, specified in [RFC4033], explicitly excludes confidentiality from its goals.) So, if an initiator starts an HTTPS communication with a recipient, while the HTTP traffic will be encrypted, the DNS exchange prior to it will not be. When other protocols will become more and more privacy-aware and secured against surveillance (e.g. [RFC8446], [I-D.ietf-quic-transport]), the use of unencrypted transports for DNS may become "the weakest link" in privacy. It is noted that there is on-going work attempting to encrypt the SNI in the TLS handshake but that this is a non-trivial problem [I-D.ietf-tls-sni-encryption].

An important specificity of the DNS traffic is that it may take a different path than the communication between the initiator and the recipient. For instance, an eavesdropper may be unable to tap the

wire between the initiator and the recipient but may have access to the wire going to the recursive resolver, or to the authoritative name servers.

The best place to tap, from an eavesdropper's point of view, is clearly between the stub resolvers and the recursive resolvers, because traffic is not limited by DNS caching.

The attack surface between the stub resolver and the rest of the world can vary widely depending upon how the end user's computer is configured. By order of increasing attack surface:

The recursive resolver can be on the end user's computer. In (currently) a small number of cases, individuals may choose to operate their own DNS resolver on their local machine. In this case, the attack surface for the connection between the stub resolver and the caching resolver is limited to that single machine.

The recursive resolver may be at the local network edge. For many/most enterprise networks and for some residential users, the caching resolver may exist on a server at the edge of the local network. In this case, the attack surface is the local network. Note that in large enterprise networks, the DNS resolver may not be located at the edge of the local network but rather at the edge of the overall enterprise network. In this case, the enterprise network could be thought of as similar to the Internet Access Provider (IAP) network referenced below.

The recursive resolver can be in the IAP premises. For most residential users and potentially other networks, the typical case is for the end user's computer to be configured (typically automatically through DHCP) with the addresses of the DNS recursive resolvers at the IAP. The attack surface for on-the-wire attacks is therefore from the end-user system across the local network and across the IAP network to the IAP's recursive resolvers.

The recursive resolver can be a public DNS service. Some machines may be configured to use public DNS resolvers such as those operated today by Google Public DNS or OpenDNS. The end user may have configured their machine to use these DNS recursive resolvers themselves -- or their IAP may have chosen to use the public DNS resolvers rather than operating their own resolvers. In this case, the attack surface is the entire public Internet between the end user's connection and the public DNS service.

2.4.2. Encrypted Transports

The use of encrypted transports directly mitigates passive surveillance of the DNS payload, however there are still some privacy attacks possible.

These are cases where user identification, fingerprinting or correlations may be possible due to the use of certain transport layers or clear text/observable features. These issues are not specific to DNS, but DNS traffic is susceptible to these attacks when using specific transports.

There are some general examples, for example, certain studies have highlighted that IP TTL or TCP Window sizes os-fingerprint [2] values can be used to fingerprint client OS's or that various techniques can be used to de-NAT DNS queries dns-de-nat [3].

The use of clear text transport options to decrease latency may also identify a user e.g. using TCP Fast Open [RFC7413].

More specifically, (since the deployment of encrypted transports is not widespread at the time of writing) users wishing to use encrypted transports for DNS may in practice be limited in the resolver services available. Given this, the choice of a user to configure a single resolver (or a fixed set of resolvers) and an encrypted transport to use in all network environments can actually serve to identify the user as one that desires privacy and can provide an added mechanism to track them as they move across network environments.

Users of encrypted transports are also highly likely to re-use sessions for multiple DNS queries to optimize performance (e.g. via DNS pipelining or HTTPS multiplexing). Certain configuration options for encrypted transports could also in principle fingerprint a user, for example session resumption, the maximum number of messages to send or a maximum connection time before closing a connections and re-opening.

Whilst there are known attacks on older versions of TLS the most recent recommendations [RFC7525] and developments [RFC8446] in this area largely mitigate those.

Traffic analysis of unpadded encrypted traffic is also possible [pitfalls-of-dns-encrption] because the sizes and timing of encrypted DNS requests and responses can be correlated to unencrypted DNS requests upstream of a recursive resolver.

2.5. In the Servers

Using the terminology of [RFC6973], the DNS servers (recursive resolvers and authoritative servers) are enablers: they facilitate communication between an initiator and a recipient without being directly in the communications path. As a result, they are often forgotten in risk analysis. But, to quote again [RFC6973], "Although [...] enablers may not generally be considered as attackers, they may all pose privacy threats (depending on the context) because they are able to observe, collect, process, and transfer privacy-relevant data." In [RFC6973] parlance, enablers become observers when they start collecting data.

Many programs exist to collect and analyze DNS data at the servers -- from the "query log" of some programs like BIND to tcpdump and more sophisticated programs like PacketQ [packetq] [packetq-list] and DNSmezzo [dnsmezzo]. The organization managing the DNS server can use this data itself, or it can be part of a surveillance program like PRISM [prism] and pass data to an outside observer.

Sometimes, this data is kept for a long time and/or distributed to third parties for research purposes [ditl] [day-at-root], security analysis, or surveillance tasks. These uses are sometimes under some sort of contract, with various limitations, for instance, on redistribution, given the sensitive nature of the data. Also, there are observation points in the network that gather DNS data and then make it accessible to third parties for research or security purposes ("passive DNS" [passive-dns]).

2.5.1. In the Recursive Resolvers

Recursive Resolvers see all the traffic since there is typically no caching before them. To summarize: your recursive resolver knows a lot about you. The resolver of a large IAP, or a large public resolver, can collect data from many users. You may get an idea of the data collected by reading the privacy policy of a big public resolver, e.g., <<https://developers.google.com/speed/public-dns/privacy>>.

2.5.1.1. Encrypted transports

Use of encrypted transports does not reduce the data available in the recursive resolver and ironically can actually expose more information about users to operators. As mentioned in Section 2.4 use of session based encrypted transports (TCP/TLS) can expose correlation data about users. Such concerns in the TCP/TLS layers apply equally to DNS-over-TLS and DoH which both use TLS as the underlying transport.

2.5.1.2. DoH vs DNS-over-TLS

The proposed specification for DoH [RFC8484] includes a Privacy Considerations section which highlights some of the differences between HTTP and DNS. As a deliberate design choice DoH inherits the privacy properties of the HTTPS stack and as a consequence introduces new privacy concerns when compared with DNS over UDP, TCP or TLS [RFC7858]. The rationale for this decision is that retaining the ability to leverage the full functionality of the HTTP ecosystem is more important than placing specific constraints on this new protocol based on privacy considerations (modulo limiting the use of HTTP cookies).

In analyzing the new issues introduced by DoH it is helpful to recognize that there exists a natural tension between

- o the wide practice in HTTP to use various headers to optimize HTTP connections, functionality and behaviour (which can facilitate user identification and tracking)
- o and the fact that the DNS payload is currently very tightly encoded and contains no standardized user identifiers.

DNS-over-TLS, for example, would normally contain no client identifiers above the TLS layer and a resolver would see only a stream of DNS query payloads originating within one or more connections from a client IP address. Whereas if DoH clients commonly include several headers in a DNS message (e.g. user-agent and accept-language) this could lead to the DoH server being able to identify the source of individual DNS requests not only to a specific end user device but to a specific application.

Additionally, depending on the client architecture, isolation of DoH queries from other HTTP traffic may or may not be feasible or desirable. Depending on the use case, isolation of DoH queries from other HTTP traffic may or may not increase privacy.

The picture for privacy considerations and user expectations for DoH with respect to what additional data may be available to the DoH server compared to DNS over UDP, TCP or TLS is complex and requires a detailed analysis for each use case. In particular the choice of HTTPS functionality vs privacy is specifically made an implementation choice in DoH and users may well have differing privacy expectations depending on the DoH use case and implementation.

At the extremes, there may be implementations that attempt to achieve parity with DNS-over-TLS from a privacy perspective at the cost of using no identifiable headers, there might be others that provide

feature rich data flows where the low-level origin of the DNS query is easily identifiable.

Privacy focussed users should be aware of the potential for additional client identifiers in DoH compared to DNS-over-TLS and may want to only use DoH implementations that provide clear guidance on what identifiers they add.

2.5.2. In the Authoritative Name Servers

Unlike what happens for recursive resolvers, observation capabilities of authoritative name servers are limited by caching; they see only the requests for which the answer was not in the cache. For aggregated statistics ("What is the percentage of LOC queries?"), this is sufficient, but it prevents an observer from seeing everything. Still, the authoritative name servers see a part of the traffic, and this subset may be sufficient to violate some privacy expectations.

Also, the end user typically has some legal/contractual link with the recursive resolver (he has chosen the IAP, or he has chosen to use a given public resolver), while having no control and perhaps no awareness of the role of the authoritative name servers and their observation abilities.

As noted before, using a local resolver or a resolver close to the machine decreases the attack surface for an on-the-wire eavesdropper. But it may decrease privacy against an observer located on an authoritative name server. This authoritative name server will see the IP address of the end client instead of the address of a big recursive resolver shared by many users.

This "protection", when using a large resolver with many clients, is no longer present if ECS [RFC7871] is used because, in this case, the authoritative name server sees the original IP address (or prefix, depending on the setup).

As of today, all the instances of one root name server, L-root, receive together around 50,000 queries per second. While most of it is "junk" (errors on the Top-Level Domain (TLD) name), it gives an idea of the amount of big data that pours into name servers. (And even "junk" can leak information; for instance, if there is a typing error in the TLD, the user will send data to a TLD that is not the usual one.)

Many domains, including TLDs, are partially hosted by third-party servers, sometimes in a different country. The contracts between the domain manager and these servers may or may not take privacy into

account. Whatever the contract, the third-party hoster may be honest or not but, in any case, it will have to follow its local laws. So, requests to a given ccTLD may go to servers managed by organizations outside of the ccTLD's country. End users may not anticipate that, when doing a security analysis.

Also, it seems (see the survey described in [aeris-dns]) that there is a strong concentration of authoritative name servers among "popular" domains (such as the Alexa Top N list). For instance, among the Alexa Top 100K, one DNS provider hosts today 10% of the domains. The ten most important DNS providers host together one third of the domains. With the control (or the ability to sniff the traffic) of a few name servers, you can gather a lot of information.

2.5.3. Rogue Servers

The previous paragraphs discussed DNS privacy, assuming that all the traffic was directed to the intended servers and that the potential attacker was purely passive. But, in reality, we can have active attackers redirecting the traffic, not to change it but just to observe it.

For instance, a rogue DHCP server, or a trusted DHCP server that has had its configuration altered by malicious parties, can direct you to a rogue recursive resolver. Most of the time, it seems to be done to divert traffic by providing lies for some domain names. But it could be used just to capture the traffic and gather information about you. Other attacks, besides using DHCP, are possible. The traffic from a DNS client to a DNS server can be intercepted along its way from originator to intended source, for instance, by transparent DNS proxies in the network that will divert the traffic intended for a legitimate DNS server. This rogue server can masquerade as the intended server and respond with data to the client. (Rogue servers that inject malicious data are possible, but it is a separate problem not relevant to privacy.) A rogue server may respond correctly for a long period of time, thereby foregoing detection. This may be done for what could be claimed to be good reasons, such as optimization or caching, but it leads to a reduction of privacy compared to if there was no attacker present. Also, malware like DNSChanger [dnschanger] can change the recursive resolver in the machine's configuration, or the routing itself can be subverted (for instance, [ripe-atlas-turkey]).

2.5.4. Authentication of servers

Both Strict mode for DNS-over-TLS and DoH require authentication of the server and therefore as long as the authentication credentials are obtained over a secure channel then using either of these

transports defeats the attack of re-directing traffic to rogue servers. Of course attacks on these secure channels are also possible, but out of the scope of this document.

2.5.5. Blocking of services

User privacy can also be at risk if there is blocking (by local network operators or more general mechanisms) of access to recursive servers that offer encrypted transports. For example active blocking of port 853 for DNS-over-TLS or of specific IP addresses (e.g. 1.1.1.1 or 2606:4700:4700::1111) could restrict the resolvers available to the client. Similarly attacks on such services e.g. DDoS could force users to switch to other services that do not offer encrypted transports for DNS.

2.6. Re-identification and Other Inferences

An observer has access not only to the data he/she directly collects but also to the results of various inferences about this data.

For instance, a user can be re-identified via DNS queries. If the adversary knows a user's identity and can watch their DNS queries for a period, then that same adversary may be able to re-identify the user solely based on their pattern of DNS queries later on regardless of the location from which the user makes those queries. For example, one study [herrmann-reidentification] found that such re-identification is possible so that "73.1% of all day-to-day links were correctly established, i.e. user u was either re-identified unambiguously (1) or the classifier correctly reported that u was not present on day t+1 any more (2)." While that study related to web browsing behavior, equally characteristic patterns may be produced even in machine-to-machine communications or without a user taking specific actions, e.g., at reboot time if a characteristic set of services are accessed by the device.

For instance, one could imagine that an intelligence agency identifies people going to a site by putting in a very long DNS name and looking for queries of a specific length. Such traffic analysis could weaken some privacy solutions.

The IAB privacy and security program also have a work in progress [RFC7624] that considers such inference-based attacks in a more general framework.

2.7. More Information

Useful background information can also be found in [tor-leak] (about the risk of privacy leak through DNS) and in a few academic papers: [yanbin-tsudik], [castillo-garcia], [fangming-hori-sakurai], and [federrath-fuchs-herrmann-piosecn].

3. Actual "Attacks"

A very quick examination of DNS traffic may lead to the false conclusion that extracting the needle from the haystack is difficult. "Interesting" primary DNS requests are mixed with useless (for the eavesdropper) secondary and tertiary requests (see the terminology in Section 1). But, in this time of "big data" processing, powerful techniques now exist to get from the raw data to what the eavesdropper is actually interested in.

Many research papers about malware detection use DNS traffic to detect "abnormal" behavior that can be traced back to the activity of malware on infected machines. Yes, this research was done for the good, but technically it is a privacy attack and it demonstrates the power of the observation of DNS traffic. See [dns-footprint], [dagon-malware], and [darkreading-dns].

Passive DNS systems [passive-dns] allow reconstruction of the data of sometimes an entire zone. They are used for many reasons -- some good, some bad. Well-known passive DNS systems keep only the DNS responses, and not the source IP address of the client, precisely for privacy reasons. Other passive DNS systems may not be so careful. And there is still the potential problems with revealing QNAMEs.

The revelations (from the Edward Snowden documents, which were leaked from the National Security Agency (NSA)) of the MORECOWBELL surveillance program [morecowbell], which uses the DNS, both passively and actively, to surreptitiously gather information about the users, is another good example showing that the lack of privacy protections in the DNS is actively exploited.

4. Legalties

To our knowledge, there are no specific privacy laws for DNS data, in any country. Interpreting general privacy laws like [data-protection-directive] or GDPR [4] applicable in the European Union in the context of DNS traffic data is not an easy task, and we do not know a court precedent here. See an interesting analysis in [sidn-entrada].

5. Security Considerations

This document is entirely about security, more precisely privacy. It just lays out the problem; it does not try to set requirements (with the choices and compromises they imply), much less define solutions. Possible solutions to the issues described here are discussed in other documents (currently too many to all be mentioned); see, for instance, 'Recommendations for DNS Privacy Operators' [I-D.ietf-dprive-bcp-op].

6. Acknowledgments

Thanks to Nathalie Boulevard and to the CENTR members for the original work that led to this document. Thanks to Ondrej Sury for the interesting discussions. Thanks to Mohsen Souissi and John Heidemann for proofreading and to Paul Hoffman, Matthijs Mekking, Marcos Sanz, Tim Wicinski, Francis Dupont, Allison Mankin, and Warren Kumari for proofreading, providing technical remarks, and making many readability improvements. Thanks to Dan York, Suzanne Woolf, Tony Finch, Stephen Farrell, Peter Koch, Simon Josefsson, and Frank Denis for good written contributions. And thanks to the IESG members for the last remarks.

7. Changelog

draft-bortzmeyer-dprive-rfc7626-bis-02

- o Update various references and fix some nits.

draft-bortzmeyer-dprive-rfc7626-bis-01

- o Update reference for dickinson-bcp-op to draft-dickinson-dprive-bcp-op

draft-borztmeyer-dprive-rfc7626-bis-00:

Initial commit. Differences to RFC7626:

- o Update many references
- o Add discussions of encrypted transports including DNS-over-TLS and DoH
- o Add section on DNS payload
- o Add section on authentication of servers
- o Add section on blocking of services

8. References

8.1. Normative References

- [RFC1034] Mockapetris, P., "Domain names - concepts and facilities", STD 13, RFC 1034, DOI 10.17487/RFC1034, November 1987, <<https://www.rfc-editor.org/info/rfc1034>>.
- [RFC1035] Mockapetris, P., "Domain names - implementation and specification", STD 13, RFC 1035, DOI 10.17487/RFC1035, November 1987, <<https://www.rfc-editor.org/info/rfc1035>>.
- [RFC6973] Cooper, A., Tschofenig, H., Aboba, B., Peterson, J., Morris, J., Hansen, M., and R. Smith, "Privacy Considerations for Internet Protocols", RFC 6973, DOI 10.17487/RFC6973, July 2013, <<https://www.rfc-editor.org/info/rfc6973>>.
- [RFC7258] Farrell, S. and H. Tschofenig, "Pervasive Monitoring Is an Attack", BCP 188, RFC 7258, DOI 10.17487/RFC7258, May 2014, <<https://www.rfc-editor.org/info/rfc7258>>.

8.2. Informative References

- [aeris-dns] Vinot, N., "Vie privée: et le DNS alors?", (In French), 2015, <<https://blog.imirhil.fr/vie-privee-et-le-dns-alors.html>>.
- [castillo-garcia] Castillo-Perez, S. and J. Garcia-Alfaro, "Anonymous Resolution of DNS Queries", 2008, <<http://deic.uab.es/~joaquin/papers/is08.pdf>>.
- [dagon-malware] Dagon, D., "Corrupted DNS Resolution Paths: The Rise of a Malicious Resolution Authority", ISC/OARC Workshop, 2007, <<https://www.dns-oarc.net/files/workshop-2007/Dagon-Resolution-corruption.pdf>>.
- [darkreading-dns] Lemos, R., "Got Malware? Three Signs Revealed In DNS Traffic", InformationWeek Dark Reading, May 2013, <<http://www.darkreading.com/analytics/security-monitoring/got-malware-three-signs-revealed-in-dns-traffic/d-d-id/1139680>>.

[data-protection-directive]

European Parliament, "Directive 95/46/EC of the European Parliament and of the council on the protection of individuals with regard to the processing of personal data and on the free movement of such data", Official Journal L 281, pp. 0031 - 0050, November 1995, <<http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=CELEX:31995L0046:EN:HTML>>.

[day-at-root]

Castro, S., Wessels, D., Fomenkov, M., and K. Claffy, "A Day at the Root of the Internet", ACM SIGCOMM Computer Communication Review, Vol. 38, Number 5, DOI 10.1145/1452335.1452341, October 2008, <<http://www.sigcomm.org/sites/default/files/ccr/papers/2008/October/1452335-1452341.pdf>>.

[denis-edns-client-subnet]

Denis, F., "Security and privacy issues of edns-client-subnet", August 2013, <<https://00f.net/2013/08/07/edns-client-subnet/>>.

[ditl]

CAIDA, "A Day in the Life of the Internet (DITL)", 2002, <<http://www.caida.org/projects/ditl/>>.

[dns-footprint]

Stoner, E., "DNS Footprint of Malware", OARC Workshop, October 2010, <<https://www.dns-oarc.net/files/workshop-201010/OARC-ers-20101012.pdf>>.

[dnshchanger]

Wikipedia, "DNSChanger", October 2013, <<https://en.wikipedia.org/w/index.php?title=DNSChanger&oldid=578749672>>.

[dnsmezzo]

Bortzmeyer, S., "DNSmezzo", 2009, <<http://www.dnsmezzo.net/>>.

[fangming-hori-sakurai]

Fangming, Z., Hori, Y., and K. Sakurai, "Analysis of Privacy Disclosure in DNS Query", 2007 International Conference on Multimedia and Ubiquitous Engineering (MUE 2007), Seoul, Korea, ISBN: 0-7695-2777-9, pp. 952-957, DOI 10.1109/MUE.2007.84, April 2007, <<http://dl.acm.org/citation.cfm?id=1262690.1262986>>.

- [federrath-fuchs-herrmann-piosecny]
Federrath, H., Fuchs, K., Herrmann, D., and C. Piosecny,
"Privacy-Preserving DNS: Analysis of Broadcast, Range
Queries and Mix-based Protection Methods", Computer
Security ESORICS 2011, Springer, page(s) 665-683,
ISBN 978-3-642-23821-5, 2011, <https://svs.informatik.uni-hamburg.de/publications/2011/2011-09-14_FFHP_PrivacyPreservingDNS_ESORICS2011.pdf>.
- [grangeia.snooping]
Grangeia, L., "DNS Cache Snooping or Snooping the Cache
for Fun and Profit", February 2004,
<http://www.msit2005.mut.ac.th/msit_media/1_2551/nete4630/materials/20080718130017Hc.pdf>.
- [herrmann-reidentification]
Herrmann, D., Gerber, C., Banse, C., and H. Federrath,
"Analyzing Characteristic Host Access Patterns for Re-
Identification of Web User Sessions",
DOI 10.1007/978-3-642-27937-9_10, 2012, <http://epub.uni-regensburg.de/21103/1/Paper_PUL_nordsec_published.pdf>.
- [I-D.ietf-dprive-bcp-op]
Dickinson, S., Overeinder, B., Rijswijk-Deij, R., and A.
Mankin, "Recommendations for DNS Privacy Service
Operators", draft-ietf-dprive-bcp-op-01 (work in
progress), December 2018.
- [I-D.ietf-quic-transport]
Iyengar, J. and M. Thomson, "QUIC: A UDP-Based Multiplexed
and Secure Transport", draft-ietf-quic-transport-17 (work
in progress), December 2018.
- [I-D.ietf-tls-sni-encryption]
Huitema, C. and E. Rescorla, "Issues and Requirements for
SNI Encryption in TLS", draft-ietf-tls-sni-encryption-04
(work in progress), November 2018.
- [morecowbell]
Grothoff, C., Wachs, M., Ermert, M., and J. Appelbaum,
"NSA's MORECOWBELL: Knell for DNS", GNUnet e.V., January
2015, <<https://gnunet.org/morecowbell>>.
- [packetq]
Dot SE, "PacketQ, a simple tool to make SQL-queries
against PCAP-files", 2011,
<<https://github.com/dotse/packetq/wiki>>.

- [packetq-list] PacketQ, "PacketQ Mailing List", <<http://lists.iis.se/mailman/listinfo/packetq>>.
- [passive-dns] Weimer, F., "Passive DNS Replication", April 2005, <<http://www.enyo.de/fw/software/dnslogger/#2>>.
- [pitfalls-of-dns-encrption] Shulman, H., "Pretty Bad Privacy:Pitfalls of DNS Encryption", <<https://www.ietf.org/mail-archive/web/dns-privacy/current/pdfWqAIUmEl47.pdf>>.
- [prism] Wikipedia, "PRISM (surveillance program)", July 2015, <[https://en.wikipedia.org/w/index.php?title=PRISM_\(surveillance_program\)&oldid=673789455](https://en.wikipedia.org/w/index.php?title=PRISM_(surveillance_program)&oldid=673789455)>.
- [RFC4033] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "DNS Security Introduction and Requirements", RFC 4033, DOI 10.17487/RFC4033, March 2005, <<https://www.rfc-editor.org/info/rfc4033>>.
- [RFC5155] Laurie, B., Sisson, G., Arends, R., and D. Blacka, "DNS Security (DNSSEC) Hashed Authenticated Denial of Existence", RFC 5155, DOI 10.17487/RFC5155, March 2008, <<https://www.rfc-editor.org/info/rfc5155>>.
- [RFC5936] Lewis, E. and A. Hoenes, Ed., "DNS Zone Transfer Protocol (AXFR)", RFC 5936, DOI 10.17487/RFC5936, June 2010, <<https://www.rfc-editor.org/info/rfc5936>>.
- [RFC6269] Ford, M., Ed., Boucadair, M., Durand, A., Levis, P., and P. Roberts, "Issues with IP Address Sharing", RFC 6269, DOI 10.17487/RFC6269, June 2011, <<https://www.rfc-editor.org/info/rfc6269>>.
- [RFC7413] Cheng, Y., Chu, J., Radhakrishnan, S., and A. Jain, "TCP Fast Open", RFC 7413, DOI 10.17487/RFC7413, December 2014, <<https://www.rfc-editor.org/info/rfc7413>>.
- [RFC7525] Sheffer, Y., Holz, R., and P. Saint-Andre, "Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", BCP 195, RFC 7525, DOI 10.17487/RFC7525, May 2015, <<https://www.rfc-editor.org/info/rfc7525>>.

- [RFC7624] Barnes, R., Schneier, B., Jennings, C., Hardie, T., Trammell, B., Huitema, C., and D. Borkmann, "Confidentiality in the Face of Pervasive Surveillance: A Threat Model and Problem Statement", RFC 7624, DOI 10.17487/RFC7624, August 2015, <<https://www.rfc-editor.org/info/rfc7624>>.
- [RFC7858] Hu, Z., Zhu, L., Heidemann, J., Mankin, A., Wessels, D., and P. Hoffman, "Specification for DNS over Transport Layer Security (TLS)", RFC 7858, DOI 10.17487/RFC7858, May 2016, <<https://www.rfc-editor.org/info/rfc7858>>.
- [RFC7871] Contavalli, C., van der Gaast, W., Lawrence, D., and W. Kumari, "Client Subnet in DNS Queries", RFC 7871, DOI 10.17487/RFC7871, May 2016, <<https://www.rfc-editor.org/info/rfc7871>>.
- [RFC7873] Eastlake 3rd, D. and M. Andrews, "Domain Name System (DNS) Cookies", RFC 7873, DOI 10.17487/RFC7873, May 2016, <<https://www.rfc-editor.org/info/rfc7873>>.
- [RFC7929] Wouters, P., "DNS-Based Authentication of Named Entities (DANE) Bindings for OpenPGP", RFC 7929, DOI 10.17487/RFC7929, August 2016, <<https://www.rfc-editor.org/info/rfc7929>>.
- [RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <<https://www.rfc-editor.org/info/rfc8446>>.
- [RFC8484] Hoffman, P. and P. McManus, "DNS Queries over HTTPS (DoH)", RFC 8484, DOI 10.17487/RFC8484, October 2018, <<https://www.rfc-editor.org/info/rfc8484>>.
- [RFC8499] Hoffman, P., Sullivan, A., and K. Fujiwara, "DNS Terminology", BCP 219, RFC 8499, DOI 10.17487/RFC8499, January 2019, <<https://www.rfc-editor.org/info/rfc8499>>.
- [ripe-atlas-turkey] Aben, E., "A RIPE Atlas View of Internet Meddling in Turkey", March 2014, <<https://labs.ripe.net/Members/emileaben/a-ripe-atlas-view-of-internet-meddling-in-turkey>>.

[sidn-entrada]

Hesselman, C., Jansen, J., Wullink, M., Vink, K., and M. Simon, "A privacy framework for 'DNS big data' applications", November 2014, <https://www.sidnlabs.nl/uploads/tx_sidnpublications/SIDN_Labs_Privacyraamwerk_Position_Paper_V1.4_ENG.pdf>.

[thomas-ditl-tcp]

Thomas, M. and D. Wessels, "An Analysis of TCP Traffic in Root Server DITL Data", DNS-OARC 2014 Fall Workshop, October 2014, <<https://indico.dns-oarc.net/event/20/session/2/contribution/15/material/slides/1.pdf>>.

[tor-leak]

Tor, "DNS leaks in Tor", 2013, <<https://www.torproject.org/docs/faq.html.en#WarningsAboutSOCKSsandDNSInformationLeaks>>.

[yanbin-tsudik]

Yanbin, L. and G. Tsudik, "Towards Plugging Privacy Leaks in the Domain Name System", October 2009, <<http://arxiv.org/abs/0910.2472>>.

8.3. URIs

[1] <https://lists.dns-oarc.net/pipermail/dns-operations/2016-January/014141.html>

[2] <http://netres.ec/?b=11B99BD>

[3] https://www.researchgate.net/publication/320322146_DNS-DNS_DNS-based_De-NAT_Scheme

[4] <https://www.eugdpr.org/the-regulation.html>

Authors' Addresses

Stephane Bortzmeyer
AFNIC
1, rue Stephenson
Montigny-le-Bretonneux
France 78180

Email: bortzmeyer+ietf@nic.fr

Sara Dickinson
Sinodun IT
Magdalen Centre
Oxford Science Park
Oxford OX4 4GA
United Kingdom

Email: sara@sinodun.com

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: September 12, 2019

M. Bretelle
Facebook
March 11, 2019

DNS-over-TLS for insecure delegations
draft-bretelle-dprive-dot-for-insecure-delegations-01

Abstract

This document describes an alternative mechanism to DANE ([RFC6698]) in order to authenticate a DNS-over-TLS (DoT [RFC7858]) authoritative server by not making DNSSEC a hard requirement, making DoT server authentication available for insecure delegations.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on September 12, 2019.

Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	2
2. Terminology	2
3. Authenticating an insecure delegation	3
3.1. Public Key Infrastructure (PKIX)	3
3.2. Subject Public Key Info (SPKI)	3
3.3. Authenticating from the parent	4
3.3.1. Example	4
4. DSPKI Resource Record	4
4.1. The DSPKI Resource Record	5
4.1.1. DSPKI RDATA Wire Format	5
5. Security Considerations	6
6. IANA Considerations	6
7. Normative References	6
Acknowledgments	7
Author's Address	7

1. Introduction

This document describes an alternative mechanism to DANE ([RFC6698]) as described in [I-D.bortzmeyer-dprive-resolver-to-auth] Section 2 extending the authentication of DNS over Transport Layer Security (DoT) [RFC7858] to insecure delegations and therefore enabling the onboarding of DoT authoritative servers without the requirement for the authorities to support DNSSEC ([RFC4033], [RFC4034], and [RFC4035]). To do so, this document introduce the Delegation Subject Public Key Info (DSPKI) resource record, its purpose, usage and format.

2. Terminology

A server that supports DNS-over-TLS is called a "DoT server" to differentiate it from a "DNS Server" (one that provides DNS service over any other protocol), likewise, a client that supports this protocol is called a "DoT client"

A secure delegation ([RFC4956] Section 2) is a signed name containing a delegation (NS RRset), and a signed DS RRset, signifying a delegation to a signed zone.

An insecure delegation ([RFC4956] Section 2) is a signed name containing a delegation (NS RRset), but lacking a DS RRset, signifying a delegation to an unsigned subzone.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP

14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. Authenticating an insecure delegation

To authenticate a DoT server of a secure delegation, it is possible to use the TLSA resource record [RFC6698] of the nameserver as described in [I-D.bortzmeyer-dprive-resolver-to-auth] Section 2, while this method is valid, the absence of support of DNSSEC for such delegations precludes the onboarding and discovery of nameservers serving those zones as DoT servers.

Without the use of DNSSEC, a delegation is not able to authenticate itself as the chain of trust cannot be followed, however other mechanisms exist to have a server authenticate itself, such as Public Key Infrastructure (PKIX [RFC6125]) , SPKI, which have their own pros and cons.

3.1. Public Key Infrastructure (PKIX)

It would be possible to authenticate the name servers of the insecure delegation using PKIX, relying on an existing trust model and trust anchors.

While simple, a single trusted Certificate Authority (CA) that breaks said trust (voluntarily or involuntarily), can issue certificate for any domains, allowing an attacker to potentially impersonate both the application and the DoT server.

Another issue that rises is that the DoT servers may use an identity which belong to the same origin as application servers, which could permit personal information (such as cookies) to be leaked to the DoT servers.

3.2. Subject Public Key Info (SPKI)

The zone owner generates his own certificate and distribute the SPKI fingerprint into the DNS.

This is in essence what, amongst other things, TLSA records solve but with the requirement for DNSSEC to be enabled and functional for the queried zone. For insecure delegations, simply advertising the SPKI fingerprint would be trivial to intercept, disable, and modify.

3.3. Authenticating from the parent

While a delegation is not secured, the DNS core infrastructure already support, for the most part, DNSSEC, meaning that if the owner of an insecure delegation could set the SPKI fingerprint in a resource record (RR) at the parent, such fingerprint could be signed and validated by the DoT client. The DoT client can then establish a TLS connection to the zone name servers and authenticate the DoT server against the fingerprint acquired earlier from the parent zone.

3.3.1. Example

example.com is an insecure delegation from .com which has set the DSPKI RRset.

A DoT client looking for records under example.com will learn from .com that example.com is delegated to

```
example.com IN 172800 NS ns1.example.com
example.com IN 172800 NS ns2.example.com
# sha256
example.com IN DSPKI (
  1 4e44f900cdeb8c769f4df97e23f8fc81
    4ac4bf45a3d9dc265a2ed925171f0b71 )
# sha512
example.com IN DSPKI (
  2 ab40ed300fd220d8c72a600069f9ceb1
    f9fd7c003117e4ef34b228da1c9d76a0
    500be99e82a0c01e7f80930a46ad28b8
    ed3d5ed2df34d822b5f56c99f45889ef
)
ns1.example.com IN 172800 AAAA 2001:db8:abcd:12:1:2:3:4
ns2.example.com IN 172800 AAAA 2001:db8:abcd:ab:1:2:3:4
```

with the accompanying signature.

The DSPKI RRset signals that the nameservers are able to support DNS-over-TLS. The DoT client can then establish a TLS connection to the DoT server and authenticate them by ensuring that the SPKI matches the one learned from the parent zone.

4. DSPKI Resource Record

There may be 0 or more DSPKI served by the parent of the delegation. 0 means that DSPKI is not supported, therefore the DoT client could try other alternatives. 1 or multiple public keys can be distributed to let the DoT client validate multiple public keys, which can be useful while doing certificate rotation or when willing to provide

different secret keys to different providers that may serve the delegated zone.

4.1. The DSPKI Resource Record

The DSPKI resource record (RR) is used to associate a DoT server public key (SPKI) with the zone it is serving.

4.1.1. DSPKI RDATA Wire Format

The RDATA of the DSPKI RR consists of a one-octet matching type field, and the DER-encoded binary structure of the SubjectPublicKeyInfo field as defined in [RFC5280].

```

      1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 3 3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
| matching type | DER-encoded SPKI field                               /
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
/                                                                    /
/                                                                    /
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

4.1.1.1. The Matching Type Field

A one-octet value, called "matching type", specifies how the SPKI is presented. The types defined in this document are:

- o 0 - Exact match on SPKI
- o 1 - SHA-256 hash of SPKI
- o 2 - SHA-512 hash of SPKI

Where the SPKI can be extracted as follow:

```
openssl x509 -in cert.pem -pubkey -noout | openssl pkey -pubin -outform der
and the SHA-256 as:
```

```
openssl x509 -in cert.pem -pubkey -noout | openssl pkey -pubin -outform der | \
openssl dgst -sha256 -binary
```

FIXME: consider

- o alternate URI to support DoT (host, port, spki), DoH (host, port, URL template), DNS-over-QUIC... would rather be an ALTNS type of record

- o CDSPKI a la CDS, CDNSKEY

5. Security Considerations

TODO Security

6. IANA Considerations

TODO: This document requires IANA actions (new RR type).

7. Normative References

[I-D.bortzmeyer-dprive-resolver-to-auth]

Bortzmeyer, S., "Encryption and authentication of the DNS resolver-to-authoritative communication", draft-bortzmeyer-dprive-resolver-to-auth-01 (work in progress), March 2018.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.

[RFC4033] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "DNS Security Introduction and Requirements", RFC 4033, DOI 10.17487/RFC4033, March 2005, <<https://www.rfc-editor.org/info/rfc4033>>.

[RFC4034] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "Resource Records for the DNS Security Extensions", RFC 4034, DOI 10.17487/RFC4034, March 2005, <<https://www.rfc-editor.org/info/rfc4034>>.

[RFC4035] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "Protocol Modifications for the DNS Security Extensions", RFC 4035, DOI 10.17487/RFC4035, March 2005, <<https://www.rfc-editor.org/info/rfc4035>>.

[RFC4956] Arends, R., Kosters, M., and D. Blacka, "DNS Security (DNSSEC) Opt-In", RFC 4956, DOI 10.17487/RFC4956, July 2007, <<https://www.rfc-editor.org/info/rfc4956>>.

[RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, DOI 10.17487/RFC5280, May 2008, <<https://www.rfc-editor.org/info/rfc5280>>.

- [RFC6125] Saint-Andre, P. and J. Hodges, "Representation and Verification of Domain-Based Application Service Identity within Internet Public Key Infrastructure Using X.509 (PKIX) Certificates in the Context of Transport Layer Security (TLS)", RFC 6125, DOI 10.17487/RFC6125, March 2011, <<https://www.rfc-editor.org/info/rfc6125>>.
- [RFC6698] Hoffman, P. and J. Schlyter, "The DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS) Protocol: TLSA", RFC 6698, DOI 10.17487/RFC6698, August 2012, <<https://www.rfc-editor.org/info/rfc6698>>.
- [RFC7250] Wouters, P., Ed., Tschofenig, H., Ed., Gilmore, J., Weiler, S., and T. Kivinen, "Using Raw Public Keys in Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", RFC 7250, DOI 10.17487/RFC7250, June 2014, <<https://www.rfc-editor.org/info/rfc7250>>.
- [RFC7858] Hu, Z., Zhu, L., Heidemann, J., Mankin, A., Wessels, D., and P. Hoffman, "Specification for DNS over Transport Layer Security (TLS)", RFC 7858, DOI 10.17487/RFC7858, May 2016, <<https://www.rfc-editor.org/info/rfc7858>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.

Acknowledgments

TODO acknowledge.

Author's Address

Emmanuel Bretelle
Facebook

Email: chantra@fb.com

dprive
Internet-Draft
Intended status: Best Current Practice
Expires: January 14, 2021

S. Dickinson
Sinodun IT
B. Overeinder
R. van Rijswijk-Deij
NLnet Labs
A. Mankin
Salesforce
July 13, 2020

Recommendations for DNS Privacy Service Operators
draft-ietf-dprive-bcp-op-14

Abstract

This document presents operational, policy, and security considerations for DNS recursive resolver operators who choose to offer DNS Privacy services. With these recommendations, the operator can make deliberate decisions regarding which services to provide, and how the decisions and alternatives impact the privacy of users.

This document also presents a non-normative framework to assist writers of a Recursive operator Privacy Statement (analogous to DNS Security Extensions (DNSSEC) Policies and DNSSEC Practice Statements described in RFC6841).

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on January 14, 2021.

Copyright Notice

Copyright (c) 2020 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction	3
2. Scope	5
3. Privacy-related documents	5
4. Terminology	6
5. Recommendations for DNS privacy services	6
5.1. On the wire between client and server	7
5.1.1. Transport recommendations	7
5.1.2. Authentication of DNS privacy services	8
5.1.3. Protocol recommendations	9
5.1.4. DNSSEC	11
5.1.5. Availability	12
5.1.6. Service options	12
5.1.7. Impact of Encryption on Monitoring by DNS Privacy Service Operators	13
5.1.8. Limitations of fronting a DNS privacy service with a pure TLS proxy	13
5.2. Data at rest on the server	14
5.2.1. Data handling	14
5.2.2. Data minimization of network traffic	15
5.2.3. IP address pseudonymization and anonymization methods	16
5.2.4. Pseudonymization, anonymization, or discarding of other correlation data	16
5.2.5. Cache snooping	17
5.3. Data sent onwards from the server	17
5.3.1. Protocol recommendations	17
5.3.2. Client query obfuscation	18
5.3.3. Data sharing	19
6. Recursive operator Privacy Statement (RPS)	20
6.1. Outline of an RPS	20
6.1.1. Policy	20
6.1.2. Practice	21
6.2. Enforcement/accountability	22
7. IANA considerations	23
8. Security considerations	23
9. Acknowledgements	23
10. Contributors	23

11. Changelog	24
12. References	27
12.1. Normative References	27
12.2. Informative References	29
Appendix A. Documents	34
A.1. Potential increases in DNS privacy	34
A.2. Potential decreases in DNS privacy	34
A.3. Related operational documents	35
Appendix B. IP address techniques	35
B.1. Categorization of techniques	36
B.2. Specific techniques	37
B.2.1. Google Analytics non-prefix filtering	37
B.2.2. dnswasher	38
B.2.3. Prefix-preserving map	38
B.2.4. Cryptographic Prefix-Preserving Pseudonymization	38
B.2.5. Top-hash Subtree-replicated Anonymization	39
B.2.6. ipcipher	39
B.2.7. Bloom filters	39
Appendix C. Current policy and privacy statements	40
Appendix D. Example RPS	40
D.1. Policy	40
D.2. Practice	43
Authors' Addresses	44

1. Introduction

The Domain Name System (DNS) is at the core of the Internet; almost every activity on the Internet starts with a DNS query (and often several). However the DNS was not originally designed with strong security or privacy mechanisms. A number of developments have taken place in recent years which aim to increase the privacy of the DNS system and these are now seeing some deployment. This latest evolution of the DNS presents new challenges to operators and this document attempts to provide an overview of considerations for privacy focused DNS services.

In recent years there has also been an increase in the availability of "public resolvers" [RFC8499] which users may prefer to use instead of the default network resolver either because they offer a specific feature (e.g., good reachability or encrypted transport) or because the network resolver lacks a specific feature (e.g., strong privacy policy or unfiltered responses). These public resolvers have tended to be at the forefront of adoption of privacy-related enhancements but it is anticipated that operators of other resolver services will follow.

Whilst protocols that encrypt DNS messages on the wire provide protection against certain attacks, the resolver operator still has

(in principle) full visibility of the query data and transport identifiers for each user. Therefore, a trust relationship (whether explicit or implicit) is assumed to exist between each user and the operator of the resolver(s) used by that user. The ability of the operator to provide a transparent, well documented, and secure privacy service will likely serve as a major differentiating factor for privacy conscious users if they make an active selection of which resolver to use.

It should also be noted that the choice of a user to configure a single resolver (or a fixed set of resolvers) and an encrypted transport to use in all network environments has both advantages and disadvantages. For example, the user has a clear expectation of which resolvers have visibility of their query data. However, this resolver/transport selection may provide an added mechanism to track them as they move across network environments. Commitments from resolver operators to minimize such tracking as users move between networks are also likely to play a role in user selection of resolvers.

More recently the global legislative landscape with regard to personal data collection, retention, and pseudonymization has seen significant activity. Providing detailed practice advice about these areas to the operator is out of scope, but Section 5.3.3 describes some mitigations of data sharing risk.

This document has two main goals:

- o To provide operational and policy guidance related to DNS over encrypted transports and to outline recommendations for data handling for operators of DNS privacy services.
- o To introduce the Recursive operator Privacy Statement (RPS) and present a framework to assist writers of an RPS. An RPS is a document that an operator should publish which outlines their operational practices and commitments with regard to privacy, thereby providing a means for clients to evaluate both the measurable and claimed privacy properties of a given DNS privacy service. The framework identifies a set of elements and specifies an outline order for them. This document does not, however, define a particular privacy statement, nor does it seek to provide legal advice as to the contents.

A desired operational impact is that all operators (both those providing resolvers within networks and those operating large public services) can demonstrate their commitment to user privacy thereby driving all DNS resolution services to a more equitable footing. Choices for users would (in this ideal world) be driven by other

factors, e.g., differing security policies or minor difference in operator policy, rather than gross disparities in privacy concerns.

Community insight [or judgment?] about operational practices can change quickly, and experience shows that a Best Current Practice (BCP) document about privacy and security is a point-in-time statement. Readers are advised to seek out any updates that apply to this document.

2. Scope

"DNS Privacy Considerations" [RFC7626] describes the general privacy issues and threats associated with the use of the DNS by Internet users and much of the threat analysis here is lifted from that document and from [RFC6973]. However this document is limited in scope to best practice considerations for the provision of DNS privacy services by servers (recursive resolvers) to clients (stub resolvers or forwarders). Choices that are made exclusively by the end user, or those for operators of authoritative nameservers are out of scope.

This document includes (but is not limited to) considerations in the following areas:

1. Data "on the wire" between a client and a server.
2. Data "at rest" on a server (e.g., in logs).
3. Data "sent onwards" from the server (either on the wire or shared with a third party).

Whilst the issues raised here are targeted at those operators who choose to offer a DNS privacy service, considerations for areas 2 and 3 could equally apply to operators who only offer DNS over unencrypted transports but who would otherwise like to align with privacy best practice.

3. Privacy-related documents

There are various documents that describe protocol changes that have the potential to either increase or decrease the privacy properties of the DNS in various ways. Note this does not imply that some documents are good or bad, better or worse, just that (for example) some features may bring functional benefits at the price of a reduction in privacy and conversely some features increase privacy with an accompanying increase in complexity. A selection of the most relevant documents are listed in Appendix A for reference.

4. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

DNS terminology is as described in [RFC8499] with one modification: we restate the clause in the original definition of Privacy-enabling DNS server in [RFC8310] to include the requirement that a DNS over (D)TLS server should also offer at least one of the credentials described in Section 8 of [RFC8310] and implement the (D)TLS profile described in Section 9 of [RFC8310].

Other Terms:

- o RPS: Recursive operator Privacy Statement, see Section 6.
- o DNS privacy service: The service that is offered via a privacy-enabling DNS server and is documented either in an informal statement of policy and practice with regard to users privacy or a formal RPS.

5. Recommendations for DNS privacy services

In the following sections we first outline the threats relevant to the specific topic and then discuss the potential actions that can be taken to mitigate them.

We describe two classes of threats:

- o Threats described in [RFC6973] 'Privacy Considerations for Internet Protocols'
 - * Privacy terminology, threats to privacy, and mitigations as described in Sections 3, 5, and 6 of [RFC6973].
- o DNS Privacy Threats
 - * These are threats to the users and operators of DNS privacy services that are not directly covered by [RFC6973]. These may be more operational in nature such as certificate management or service availability issues.

We describe three classes of actions that operators of DNS privacy services can take:

- o Threat mitigation for well understood and documented privacy threats to the users of the service and in some cases to the operators of the service.
- o Optimization of privacy services from an operational or management perspective.
- o Additional options that could further enhance the privacy and usability of the service.

This document does not specify policy - only best practice, however for DNS Privacy services to be considered compliant with these best practice guidelines they SHOULD implement (where appropriate) all:

- o Threat mitigations to be minimally compliant.
- o Optimizations to be moderately compliant.
- o Additional options to be maximally compliant.

The rest of this document does not use normative language but instead refers only to the three differing classes of action which correspond to the three named levels of compliance stated above. However, compliance (to the indicated level) remains a normative requirement.

5.1. On the wire between client and server

In this section we consider both data on the wire and the service provided to the client.

5.1.1. Transport recommendations

[RFC6973] Threats:

- o Surveillance:
 - * Passive surveillance of traffic on the wire

DNS Privacy Threats:

- o Active injection of spurious data or traffic.

Mitigations:

A DNS privacy service can mitigate these threats by providing service over one or more of the following transports

- o DNS over TLS (DoT) [RFC7858] and [RFC8310].

- o DNS over HTTPS (DoH) [RFC8484].

It is noted that a DNS privacy service can also be provided over DNS over DTLS [RFC8094], however this is an Experimental specification and there are no known implementations at the time of writing.

It is also noted that DNS privacy service might be provided over IPsec, DNSCrypt, or VPNs. However, there are no specific RFCs that cover the use of these transports for DNS and any discussion of best practice for providing such a service is out of scope for this document.

Whilst encryption of DNS traffic can protect against active injection on the paths traversed by the encrypted connection this does not diminish the need for DNSSEC, see Section 5.1.4.

5.1.2. Authentication of DNS privacy services

[RFC6973] Threats:

- o Surveillance:

- * Active attacks on client resolver configuration

Mitigations:

DNS privacy services should ensure clients can authenticate the server. Note that this, in effect, commits the DNS privacy service to a public identity users will trust.

When using DoT, clients that select a 'Strict Privacy' usage profile [RFC8310] (to mitigate the threat of active attack on the client) require the ability to authenticate the DNS server. To enable this, DNS privacy services that offer DNS over TLS need to provide credentials that will be accepted by the client's trust model, in the form of either X.509 certificates [RFC5280] or Subject Public Key Info (SPKI) pin sets [RFC8310].

When offering DoH [RFC8484], HTTPS requires authentication of the server as part of the protocol.

Server operators should also follow the best practices with regard to certificate revocation as described in [RFC7525].

5.1.2.1. Certificate management

Anecdotal evidence to date highlights the management of certificates as one of the more challenging aspects for operators of traditional DNS resolvers that choose to additionally provide a DNS privacy service as management of such credentials is new to those DNS operators.

It is noted that SPKI pin set management is described in [RFC7858] but that key pinning mechanisms in general have fallen out of favor operationally for various reasons such as the logistical overhead of rolling keys.

DNS Privacy Threats:

- o Invalid certificates, resulting in an unavailable service which might force a user to fallback to cleartext.
- o Mis-identification of a server by a client e.g., typos in DoH URL templates [RFC8484] or authentication domain names [RFC8310] which accidentally direct clients to attacker controlled servers.

Mitigations:

It is recommended that operators:

- o Follow the guidance in Section 6.5 of [RFC7525] with regards to certificate revocation.
- o Automate the generation, publication, and renewal of certificates. For example, ACME [RFC8555] provides a mechanism to actively manage certificates through automation and has been implemented by a number of certificate authorities.
- o Monitor certificates to prevent accidental expiration of certificates.
- o Choose a short, memorable authentication domain name for the service.

5.1.3. Protocol recommendations

5.1.3.1. DoT

DNS Privacy Threats:

- o Known attacks on TLS such as those described in [RFC7457].

- o Traffic analysis, for example: [Pitfalls-of-DNS-Encryption].
- o Potential for client tracking via transport identifiers.
- o Blocking of well known ports (e.g., 853 for DoT).

Mitigations:

In the case of DoT, TLS profiles from Section 9 of [RFC8310] and the Countermeasures to DNS Traffic Analysis from section 11.1 of [RFC8310] provide strong mitigations. This includes but is not limited to:

- o Adhering to [RFC7525].
- o Implementing only (D)TLS 1.2 or later as specified in [RFC8310].
- o Implementing EDNS(0) Padding [RFC7830] using the guidelines in [RFC8467] or a successor specification.
- o Servers should not degrade in any way the query service level provided to clients that do not use any form of session resumption mechanism, such as TLS session resumption [RFC5077] with TLS 1.2, section 2.2 of [RFC8446], or Domain Name System (DNS) Cookies [RFC7873].
- o A DoT privacy service on both port 853 and 443. If the operator deploys DoH on the same IP address this requires the use of the 'dot' ALPN value [dot-ALPN].

Optimizations:

- o Concurrent processing of pipelined queries, returning responses as soon as available, potentially out of order as specified in [RFC7766]. This is often called 'OOOR' - out-of-order responses (providing processing performance similar to HTTP multiplexing).
- o Management of TLS connections to optimize performance for clients using [RFC7766] and EDNS(0) Keepalive [RFC7828]

Additional Options:

Management of TLS connections to optimize performance for clients using DNS Stateful Operations [RFC8490].

5.1.3.2. DoH

DNS Privacy Threats:

- o Known attacks on TLS such as those described in [RFC7457].
- o Traffic analysis, for example: [DNS-Privacy-not-so-private].
- o Potential for client tracking via transport identifiers.

Mitigations:

- o Clients must be able to forgo the use of HTTP Cookies [RFC6265] and still use the service.
- o Use of HTTP/2 padding and/or EDNS(0) padding as described in Section 9 of [RFC8484]
- o Clients should not be required to include any headers beyond the absolute minimum to obtain service from a DoH server. (See Section 6.1 of [I-D.ietf-httpbis-bcp56bis].)

5.1.4. DNSSEC

DNS Privacy Threats:

- o Users may be directed to bogus IP addresses which, depending on the application, protocol and authentication method, might lead users to reveal personal information to attackers. One example is a website that doesn't use TLS or its TLS authentication can somehow be subverted.

Mitigations:

- o All DNS privacy services must offer a DNS privacy service that performs Domain Name System Security Extensions (DNSSEC) validation. In addition they must be able to provide the DNSSEC RRs to the client so that it can perform its own validation.

The addition of encryption to DNS does not remove the need for DNSSEC [RFC4033] - they are independent and fully compatible protocols, each solving different problems. The use of one does not diminish the need nor the usefulness of the other.

While the use of an authenticated and encrypted transport protects origin authentication and data integrity between a client and a DNS privacy service it provides no proof (for a non-validating client) that the data provided by the DNS privacy service was actually DNSSEC

authenticated. As with cleartext DNS the user is still solely trusting the AD bit (if present) set by the resolver.

It should also be noted that the use of an encrypted transport for DNS actually solves many of the practical issues encountered by DNS validating clients e.g. interference by middleboxes with cleartext DNS payloads is completely avoided. In this sense a validating client that uses a DNS privacy service which supports DNSSEC has a far simpler task in terms of DNSSEC Roadblock avoidance [RFC8027].

5.1.5. Availability

DNS Privacy Threats:

- o A failed DNS privacy service could force the user to switch providers, fallback to cleartext or accept no DNS service for the outage.

Mitigations:

A DNS privacy service should strive to engineer encrypted services to the same availability level as any unencrypted services they provide. Particular care should be taken to protect DNS privacy services against denial-of-service attacks, as experience has shown that unavailability of DNS resolving because of attacks is a significant motivation for users to switch services. See, for example Section IV-C of [Passive-Observations-of-a-Large-DNS].

Techniques such as those described in Section 10 of [RFC7766] can be of use to operators to defend against such attacks.

5.1.6. Service options

DNS Privacy Threats:

- o Unfairly disadvantaging users of the privacy service with respect to the services available. This could force the user to switch providers, fallback to cleartext or accept no DNS service for the outage.

Mitigations:

A DNS privacy service should deliver the same level of service as offered on un-encrypted channels in terms of options such as filtering (or lack thereof), DNSSEC validation, etc.

5.1.7. Impact of Encryption on Monitoring by DNS Privacy Service Operators

DNS Privacy Threats:

- o Increased use of encryption can impact DNS privacy service operator ability to monitor traffic and therefore manage their DNS servers [RFC8404].

Many monitoring solutions for DNS traffic rely on the plain text nature of this traffic and work by intercepting traffic on the wire, either using a separate view on the connection between clients and the resolver, or as a separate process on the resolver system that inspects network traffic. Such solutions will no longer function when traffic between clients and resolvers is encrypted. Many DNS privacy service operators still have need to inspect DNS traffic, e.g., to monitor for network security threats. Operators may therefore need to invest in alternative means of monitoring that relies on either the resolver software directly, or exporting DNS traffic from the resolver using e.g., [dnstap].

Optimization:

When implementing alternative means for traffic monitoring, operators of a DNS privacy service should consider using privacy conscious means to do so (see section Section 5.2 for more details on data handling and also the discussion on the use of Bloom Filters in Appendix B.

5.1.8. Limitations of fronting a DNS privacy service with a pure TLS proxy

DNS Privacy Threats:

- o Limited ability to manage or monitor incoming connections using DNS specific techniques.
- o Misconfiguration (e.g., of the target server address in the proxy configuration) could lead to data leakage if the proxy to target server path is not encrypted.

Optimization:

Some operators may choose to implement DoT using a TLS proxy (e.g. [nginx], [haproxy], or [stunnel]) in front of a DNS nameserver because of proven robustness and capacity when handling large numbers of client connections, load balancing capabilities and good tooling. Currently, however, because such proxies typically have no specific

handling of DNS as a protocol over TLS or DTLS using them can restrict traffic management at the proxy layer and at the DNS server. For example, all traffic received by a nameserver behind such a proxy will appear to originate from the proxy and DNS techniques such as ACLs, RRL, or DNS64 will be hard or impossible to implement in the nameserver.

Operators may choose to use a DNS aware proxy such as [dnsdist] which offers custom options (similar to that proposed in [I-D.bellis-dnsop-xpf]) to add source information to packets to address this shortcoming. It should be noted that such options potentially significantly increase the leaked information in the event of a misconfiguration.

5.2. Data at rest on the server

5.2.1. Data handling

[RFC6973] Threats:

- o Surveillance.
- o Stored data compromise.
- o Correlation.
- o Identification.
- o Secondary use.
- o Disclosure.

Other Threats

- o Contravention of legal requirements not to process user data.

Mitigations:

The following are recommendations relating to common activities for DNS service operators and in all cases data retention should be minimized or completely avoided if possible for DNS privacy services. If data is retained it should be encrypted and either aggregated, pseudonymized, or anonymized whenever possible. In general the principle of data minimization described in [RFC6973] should be applied.

- o Transient data (e.g., that is used for real time monitoring and threat analysis which might be held only in memory) should be

retained for the shortest possible period deemed operationally feasible.

- o The retention period of DNS traffic logs should be only those required to sustain operation of the service and, to the extent that such exists, meet regulatory requirements.
- o DNS privacy services should not track users except for the particular purpose of detecting and remedying technically malicious (e.g., DoS) or anomalous use of the service.
- o Data access should be minimized to only those personnel who require access to perform operational duties. It should also be limited to anonymized or pseudonymized data where operationally feasible, with access to full logs (if any are held) only permitted when necessary.

Optimizations:

- o Consider use of full disk encryption for logs and data capture storage.

5.2.2. Data minimization of network traffic

Data minimization refers to collecting, using, disclosing, and storing the minimal data necessary to perform a task, and this can be achieved by removing or obfuscating privacy-sensitive information in network traffic logs. This is typically personal data, or data that can be used to link a record to an individual, but may also include revealing other confidential information, for example on the structure of an internal corporate network.

The problem of effectively ensuring that DNS traffic logs contain no or minimal privacy-sensitive information is not one that currently has a generally agreed solution or any standards to inform this discussion. This section presents an overview of current techniques to simply provide reference on the current status of this work.

Research into data minimization techniques (and particularly IP address pseudonymization/anonymization) was sparked in the late 1990s/early 2000s, partly driven by the desire to share significant corpuses of traffic captures for research purposes. Several techniques reflecting different requirements in this area and different performance/resource tradeoffs emerged over the course of the decade. Developments over the last decade have been both a blessing and a curse; the large increase in size between an IPv4 and an IPv6 address, for example, renders some techniques impractical, but also makes available a much larger amount of input entropy, the

better to resist brute force re-identification attacks that have grown in practicality over the period.

Techniques employed may be broadly categorized as either anonymization or pseudonymization. The following discussion uses the definitions from [RFC6973] Section 3, with additional observations from [van-Dijkhuizen-et-al.]

- o Anonymization. To enable anonymity of an individual, there must exist a set of individuals that appear to have the same attribute(s) as the individual. To the attacker or the observer, these individuals must appear indistinguishable from each other.
- o Pseudonymization. The true identity is deterministically replaced with an alternate identity (a pseudonym). When the pseudonymization schema is known, the process can be reversed, so the original identity becomes known again.

In practice there is a fine line between the two; for example, how to categorize a deterministic algorithm for data minimization of IP addresses that produces a group of pseudonyms for a single given address.

5.2.3. IP address pseudonymization and anonymization methods

A major privacy risk in DNS is connecting DNS queries to an individual and the major vector for this in DNS traffic is the client IP address.

There is active discussion in the space of effective pseudonymization of IP addresses in DNS traffic logs, however there seems to be no single solution that is widely recognized as suitable for all or most use cases. There are also as yet no standards for this that are unencumbered by patents.

Appendix B provides a more detailed survey of various techniques employed or under development in 2019.

5.2.4. Pseudonymization, anonymization, or discarding of other correlation data

DNS Privacy Threats:

- o Fingerprinting of the client OS via various means including: IP TTL/Hoplimit, TCP parameters (e.g., window size, ECN support, SACK), OS specific DNS query patterns (e.g., for network connectivity, captive portal detection, or OS specific updates).

- o Fingerprinting of the client application or TLS library by, e.g., HTTP headers (e.g., User-Agent, Accept, Accept-Encoding), TLS version/Cipher suite combinations, or other connection parameters.
- o Correlation of queries on multiple TCP sessions originating from the same IP address.
- o Correlating of queries on multiple TLS sessions originating from the same client, including via session resumption mechanisms.
- o Resolvers might receive client identifiers, e.g., MAC addresses in EDNS(0) options - some Customer-premises equipment (CPE) devices are known to add them [MAC-address-EDNS].

Mitigations:

- o Data minimization or discarding of such correlation data.

5.2.5. Cache snooping

[RFC6973] Threats:

- o Surveillance:
 - * Profiling of client queries by malicious third parties.

Mitigations:

- o See [ISC-Knowledge-database-on-cache-snooping] for an example discussion on defending against cache snooping. Options proposed include limiting access to a server and limiting non-recursive queries.

5.3. Data sent onwards from the server

In this section we consider both data sent on the wire in upstream queries and data shared with third parties.

5.3.1. Protocol recommendations

[RFC6973] Threats:

- o Surveillance:
 - * Transmission of identifying data upstream.

Mitigations:

As specified in [RFC8310] for DoT but applicable to any DNS Privacy services the server should:

- o Implement QNAME minimization [RFC7816].
- o Honor a SOURCE PREFIX-LENGTH set to 0 in a query containing the EDNS(0) Client Subnet (ECS) option ([RFC7871] Section 7.1.2).

Optimizations:

- o As per Section 2 of [RFC7871] the server should either:
 - * not use the ECS option in upstream queries at all, or
 - * offer alternative services, one that sends ECS and one that does not.

If operators do offer a service that sends the ECS options upstream they should use the shortest prefix that is operationally feasible and ideally use a policy of allowlisting upstream servers to send ECS to in order to reduce data leakage. Operators should make clear in any policy statement what prefix length they actually send and the specific policy used.

Allowlisting has the benefit that not only does the operator know which upstream servers can use ECS but also allows the operator to decide which upstream servers apply privacy policies that the operator is happy with. However some operators consider allowlisting to incur significant operational overhead compared to dynamic detection of ECS support on authoritative servers.

Additional options:

- o Aggressive Use of DNSSEC-Validated Cache [RFC8198] and [RFC8020] (NXDOMAIN: There Really Is Nothing Underneath) to reduce the number of queries to authoritative servers to increase privacy.
- o Run a copy of the root zone on loopback [RFC8806] to avoid making queries to the root servers that might leak information.

5.3.2. Client query obfuscation

Additional options:

Since queries from recursive resolvers to authoritative servers are performed using cleartext (at the time of writing), resolver services need to consider the extent to which they may be directly leaking information about their client community via these upstream queries

and what they can do to mitigate this further. Note, that even when all the relevant techniques described above are employed there may still be attacks possible, e.g. [Pitfalls-of-DNS-Encryption]. For example, a resolver with a very small community of users risks exposing data in this way and ought to obfuscate this traffic by mixing it with 'generated' traffic to make client characterization harder. The resolver could also employ aggressive pre-fetch techniques as a further measure to counter traffic analysis.

At the time of writing there are no standardized or widely recognized techniques to perform such obfuscation or bulk pre-fetches.

Another technique that particularly small operators may consider is forwarding local traffic to a larger resolver (with a privacy policy that aligns with their own practices) over an encrypted protocol so that the upstream queries are obfuscated among those of the large resolver.

5.3.3. Data sharing

[RFC6973] Threats:

- o Surveillance.
- o Stored data compromise.
- o Correlation.
- o Identification.
- o Secondary use.
- o Disclosure.

DNS Privacy Threats:

- o Contravention of legal requirements not to process user data.

Mitigations:

Operators should not share identifiable data with third-parties.

If operators choose to share identifiable data with third-parties in specific circumstance they should publish the terms under which data is shared.

Operators should consider including specific guidelines for the collection of aggregated and/or anonymized data for research

purposes, within or outside of their own organization. This can benefit not only the operator (through inclusion in novel research) but also the wider Internet community. See the policy published by SURFnet [SURFnet-policy] on data sharing for research as an example.

6. Recursive operator Privacy Statement (RPS)

To be compliant with this Best Common Practices document, a DNS recursive operator SHOULD publish a Recursive operator Privacy Statement (RPS). Adopting the outline, and including the headings in the order provided, is a benefit to persons comparing RPSs from multiple operators.

Appendix C provides a comparison of some existing policy and privacy statements.

6.1. Outline of an RPS

The contents of Section 6.1.1 and Section 6.1.2 are non-normative, other than the order of the headings. Material under each topic is present to assist the operator developing their own RPS and:

- o Relates only to matters around to the technical operation of DNS privacy services, and not on any other matters.
- o Does not attempt to offer an exhaustive list for the contents of an RPS.
- o Is not intended to form the basis of any legal/compliance documentation.

Appendix D provides an example (also non-normative) of an RPS statement for a specific operator scenario.

6.1.1. Policy

1. Treatment of IP addresses. Make an explicit statement that IP addresses are treated as personal data.
2. Data collection and sharing. Specify clearly what data (including IP addresses) is:
 - * Collected and retained by the operator, and for what period it is retained.
 - * Shared with partners.
 - * Shared, sold, or rented to third-parties.

and in each case whether it is aggregated, pseudonymized, or anonymized and the conditions of data transfer. Where possible provide details of the techniques used for the above data minimizations.

3. Exceptions. Specify any exceptions to the above, for example, technically malicious or anomalous behavior.
4. Associated entities. Declare and explicitly enumerate any partners, third-party affiliations, or sources of funding.
5. Correlation. Whether user DNS data is correlated or combined with any other personal information held by the operator.
6. Result filtering. This section should explain whether the operator filters, edits or alters in any way the replies that it receives from the authoritative servers for each DNS zone, before forwarding them to the clients. For each category listed below, the operator should also specify how the filtering lists are created and managed, whether it employs any third-party sources for such lists, and which ones.
 - * Specify if any replies are being filtered out or altered for network and computer security reasons (e.g., preventing connections to malware-spreading websites or botnet control servers).
 - * Specify if any replies are being filtered out or altered for mandatory legal reasons, due to applicable legislation or binding orders by courts and other public authorities.
 - * Specify if any replies are being filtered out or altered for voluntary legal reasons, due to an internal policy by the operator aiming at reducing potential legal risks.
 - * Specify if any replies are being filtered out or altered for any other reason, including commercial ones.

6.1.2. Practice

[NOTE FOR RFC EDITOR: Please update this section to use letters for the sub-bullet points instead of numbers. This was not done during review because the markdown tool used to write the document did not support it.]

Communicate the current operational practices of the service.

1. Deviations. Specify any temporary or permanent deviations from the policy for operational reasons.
 2. Client facing capabilities. With reference to each subsection of Section 5.1 provide specific details of which capabilities (transport, DNSSEC, padding, etc.) are provided on which client facing addresses/port combination or DoH URI template. For Section 5.1.2, clearly specify which specific authentication mechanisms are supported for each endpoint that offers DoT:
 1. The authentication domain name to be used (if any).
 2. The SPKI pin sets to be used (if any) and policy for rolling keys.
 3. Upstream capabilities. With reference to section Section 5.3 provide specific details of which capabilities are provided upstream for data sent to authoritative servers.
 4. Support. Provide contact/support information for the service.
 5. Data Processing. This section can optionally communicate links to and the high level contents of any separate statements the operator has published which cover applicable data processing legislation or agreements with regard to the location(s) of service provision.
- 6.2. Enforcement/accountability

Transparency reports may help with building user trust that operators adhere to their policies and practices.

Independent monitoring or analysis could be performed where possible of:

- o ECS, QNAME minimization, EDNS(0) padding, etc.
- o Filtering.
- o Uptime.

This is by analogy with several TLS or website analysis tools that are currently available e.g., [SSL-Labs] or [Internet.nl].

Additionally operators could choose to engage the services of a third party auditor to verify their compliance with their published RPS.

7. IANA considerations

None

8. Security considerations

Security considerations for DNS over TCP are given in [RFC7766], many of which are generally applicable to session based DNS. Guidance on operational requirements for DNS over TCP are also available in [I-D.dnsop-dns-tcp-requirements]. Security considerations for DoT are given in [RFC7858] and [RFC8310], those for DoH in [RFC8484].

Security considerations for DNSSEC are given in [RFC4033], [RFC4034] and [RFC4035].

9. Acknowledgements

Many thanks to Amelia Andersdotter for a very thorough review of the first draft of this document and Stephen Farrell for a thorough review at WGLC and for suggesting the inclusion of an example RPS. Thanks to John Todd for discussions on this topic, and to Stephane Bortzmeyer, Puneet Sood and Vittorio Bertola for review. Thanks to Daniel Kahn Gillmor, Barry Green, Paul Hoffman, Dan York, Jon Reed, Lorenzo Colitti for comments at the mic. Thanks to Loganaden Velvindron for useful updates to the text.

Sara Dickinson thanks the Open Technology Fund for a grant to support the work on this document.

10. Contributors

The below individuals contributed significantly to the document:

John Dickinson
Sinodun Internet Technologies
Magdalen Centre
Oxford Science Park
Oxford OX4 4GA
United Kingdom

Jim Hague
Sinodun Internet Technologies
Magdalen Centre
Oxford Science Park
Oxford OX4 4GA
United Kingdom

11. Changelog

draft-ietf-dprive-bcp-op-13

- o Minor edits

draft-ietf-dprive-bcp-op-12

- o Change DROP to RPS throughout

draft-ietf-dprive-bcp-op-11

- o Improve text around use of normative language
- o Fix section 5.1.3.2 bullets
- o Improve text in 6.1.2. item 2.
- o Rework text of 6.1.2. item 5 and update example DROP
- o Various editorial improvements

draft-ietf-dprive-bcp-op-10

- o Remove direct references to draft-ietf-dprive-rfc7626-bis, instead have one general reference RFC7626
- o Clarify that the DROP statement outline is non-normative and add some further qualifications about content
- o Update wording on data sharing to remove explicit discussion of consent
- o Move table in section 5.2.3 to an appendix
- o Move section 6.2 to an appendix
- o Corrections to references, typos and editorial updates from initial IESG comments.

draft-ietf-dprive-bcp-op-09

- o Fix references so they match the correct section numbers in draft-ietf-dprive-rfc7626-bis-05

draft-ietf-dprive-bcp-op-08

- o Address IETF Last call comments.

draft-ietf-dprive-bcp-op-07

- o Editorial changes following AD review.
- o Change all URIs to Informational References.

draft-ietf-dprive-bcp-op-06

- o Final minor changes from second WGLC.

draft-ietf-dprive-bcp-op-05

- o Remove some text on consent:
 - * Paragraph 2 in section 5.3.3
 - * Item 6 in the DROP Practice statement (and example)
- o Remove .onion and TLSA options
- o Include ACME as a reference for certificate management
- o Update text on session resumption usage
- o Update section 5.2.4 on client fingerprinting

draft-ietf-dprive-bcp-op-04

- o Change DPPP to DROP (DNS Recursive Operator Privacy) statement
- o Update structure of DROP slightly
- o Add example DROP statement
- o Add text about restricting access to full logs
- o Move table in section 5.2.3 from SVG to inline table
- o Fix many editorial and reference nits

draft-ietf-dprive-bcp-op-03

- o Add paragraph about operational impact
- o Move DNSSEC requirement out of the Appendix into main text as a privacy threat that should be mitigated
- o Add TLS version/Cipher suite as tracking threat

- o Add reference to Mozilla TRR policy
 - o Remove several TODOs and QUESTIONS.
- draft-ietf-dprive-bcp-op-02
- o Change 'open resolver' for 'public resolver'
 - o Minor editorial changes
 - o Remove recommendation to run a separate TLS 1.3 service
 - o Move TLSA to purely a optimization in Section 5.2.1
 - o Update reference on minimal DoH headers.
 - o Add reference on user switching provider after service issues in Section 5.1.4
 - o Add text in Section 5.1.6 on impact on operators.
 - o Add text on additional threat to TLS proxy use (Section 5.1.7)
 - o Add reference in Section 5.3.1 on example policies.
- draft-ietf-dprive-bcp-op-01
- o Many minor editorial fixes
 - o Update DoH reference to RFC8484 and add more text on DoH
 - o Split threat descriptions into ones directly referencing RFC6973 and other DNS Privacy threats
 - o Improve threat descriptions throughout
 - o Remove reference to the DNSSEC TLS Chain Extension draft until new version submitted.
 - o Clarify use of allowlisting for ECS
 - o Re-structure the DPPPS, add Result filtering section.
 - o Remove the direct inclusion of privacy policy comparison, now just reference dnsprivacy.org and an example of such work.
 - o Add an appendix briefly discussing DNSSEC

- o Update affiliation of 1 author

draft-ietf-dprive-bcp-op-00

- o Initial commit of re-named document after adoption to replace draft-dickinson-dprive-bcp-op-01

12. References

12.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC4033] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "DNS Security Introduction and Requirements", RFC 4033, DOI 10.17487/RFC4033, March 2005, <<https://www.rfc-editor.org/info/rfc4033>>.
- [RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, DOI 10.17487/RFC5280, May 2008, <<https://www.rfc-editor.org/info/rfc5280>>.
- [RFC6973] Cooper, A., Tschofenig, H., Aboba, B., Peterson, J., Morris, J., Hansen, M., and R. Smith, "Privacy Considerations for Internet Protocols", RFC 6973, DOI 10.17487/RFC6973, July 2013, <<https://www.rfc-editor.org/info/rfc6973>>.
- [RFC7457] Sheffer, Y., Holz, R., and P. Saint-Andre, "Summarizing Known Attacks on Transport Layer Security (TLS) and Datagram TLS (DTLS)", RFC 7457, DOI 10.17487/RFC7457, February 2015, <<https://www.rfc-editor.org/info/rfc7457>>.
- [RFC7525] Sheffer, Y., Holz, R., and P. Saint-Andre, "Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)", BCP 195, RFC 7525, DOI 10.17487/RFC7525, May 2015, <<https://www.rfc-editor.org/info/rfc7525>>.
- [RFC7766] Dickinson, J., Dickinson, S., Bellis, R., Mankin, A., and D. Wessels, "DNS Transport over TCP - Implementation Requirements", RFC 7766, DOI 10.17487/RFC7766, March 2016, <<https://www.rfc-editor.org/info/rfc7766>>.

- [RFC7816] Bortzmeyer, S., "DNS Query Name Minimisation to Improve Privacy", RFC 7816, DOI 10.17487/RFC7816, March 2016, <<https://www.rfc-editor.org/info/rfc7816>>.
- [RFC7828] Wouters, P., Abley, J., Dickinson, S., and R. Bellis, "The edns-tcp-keepalive EDNS0 Option", RFC 7828, DOI 10.17487/RFC7828, April 2016, <<https://www.rfc-editor.org/info/rfc7828>>.
- [RFC7830] Mayrhofer, A., "The EDNS(0) Padding Option", RFC 7830, DOI 10.17487/RFC7830, May 2016, <<https://www.rfc-editor.org/info/rfc7830>>.
- [RFC7858] Hu, Z., Zhu, L., Heidemann, J., Mankin, A., Wessels, D., and P. Hoffman, "Specification for DNS over Transport Layer Security (TLS)", RFC 7858, DOI 10.17487/RFC7858, May 2016, <<https://www.rfc-editor.org/info/rfc7858>>.
- [RFC7871] Contavalli, C., van der Gaast, W., Lawrence, D., and W. Kumari, "Client Subnet in DNS Queries", RFC 7871, DOI 10.17487/RFC7871, May 2016, <<https://www.rfc-editor.org/info/rfc7871>>.
- [RFC8020] Bortzmeyer, S. and S. Huque, "NXDOMAIN: There Really Is Nothing Underneath", RFC 8020, DOI 10.17487/RFC8020, November 2016, <<https://www.rfc-editor.org/info/rfc8020>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.
- [RFC8198] Fujiwara, K., Kato, A., and W. Kumari, "Aggressive Use of DNSSEC-Validated Cache", RFC 8198, DOI 10.17487/RFC8198, July 2017, <<https://www.rfc-editor.org/info/rfc8198>>.
- [RFC8310] Dickinson, S., Gillmor, D., and T. Reddy, "Usage Profiles for DNS over TLS and DNS over DTLS", RFC 8310, DOI 10.17487/RFC8310, March 2018, <<https://www.rfc-editor.org/info/rfc8310>>.
- [RFC8467] Mayrhofer, A., "Padding Policies for Extension Mechanisms for DNS (EDNS(0))", RFC 8467, DOI 10.17487/RFC8467, October 2018, <<https://www.rfc-editor.org/info/rfc8467>>.
- [RFC8484] Hoffman, P. and P. McManus, "DNS Queries over HTTPS (DoH)", RFC 8484, DOI 10.17487/RFC8484, October 2018, <<https://www.rfc-editor.org/info/rfc8484>>.

- [RFC8490] Bellis, R., Cheshire, S., Dickinson, J., Dickinson, S., Lemon, T., and T. Pusateri, "DNS Stateful Operations", RFC 8490, DOI 10.17487/RFC8490, March 2019, <<https://www.rfc-editor.org/info/rfc8490>>.
- [RFC8499] Hoffman, P., Sullivan, A., and K. Fujiwara, "DNS Terminology", BCP 219, RFC 8499, DOI 10.17487/RFC8499, January 2019, <<https://www.rfc-editor.org/info/rfc8499>>.
- [RFC8806] Kumari, W. 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>>.

12.2. Informative References

- [Bloom-filter]
van Rijswijk-Deij, R., Rijnders, G., Bomhoff, M., and L. Allodi, "Privacy-Conscious Threat Intelligence Using DNSBLOOM", 2019, <<http://dl.ifip.org/db/conf/im/im2019/189282.pdf>>.
- [Brenker-and-Arnes]
Brekne, T. and A. Arnes, "CIRCUMVENTING IP-ADDRESS PSEUDONYMIZATION", 2005, <<https://pdfs.semanticscholar.org/7b34/12c951cebe71cd2cddac5fda164fb2138a44.pdf>>.
- [Crypto-PAn]
CESNET, "Crypto-PAn", 2015, <<https://github.com/CESNET/ipfixcol/tree/master/base/src/intermediate/anonymization/Crypto-PAn>>.
- [DNS-Privacy-not-so-private]
Silby, S., Juarez, M., Vallina-Rodriguez, N., and C. Troncosol, "DNS Privacy not so private: the traffic analysis perspective.", 2019, <<https://petsymposium.org/2018/files/hotpets/4-siby.pdf>>.
- [dnsmist] PowerDNS, "dnsmist Overview", 2019, <<https://dnsmist.org>>.
- [dnstap] dnstap.info, "DNSTAP", 2019, <<http://dnstap.info>>.
- [DoH-resolver-policy]
Mozilla, "Security/DOH-resolver-policy", 2019, <<https://wiki.mozilla.org/Security/DOH-resolver-policy>>.

- [dot-ALPN] IANA (iana.org), "TLS Application-Layer Protocol Negotiation (ALPN) Protocol IDs", 2020, <<https://www.iana.org/assignments/tls-extensiontype-values/tls-extensiontype-values.xhtml#alpn-protocol-ids>>.
- [Geolocation-Impact-Assesment] Conversion Works, "Anonymize IP Geolocation Accuracy Impact Assessment", 2017, <<https://support.google.com/analytics/answer/2763052?hl=en>>.
- [haproxy] haproxy.org, "HAPROXY", 2019, <<https://www.haproxy.org/>>.
- [Harvan] Harvan, M., "Prefix- and Lexicographical-order-preserving IP Address Anonymization", 2006, <http://mharvan.net/talks/noms-ip_anon.pdf>.
- [I-D.bellis-dnsop-xpf] Bellis, R., Dijk, P., and R. Gacogne, "DNS X-Proxied-For", draft-bellis-dnsop-xpf-04 (work in progress), March 2018.
- [I-D.ietf-dnsop-dns-tcp-requirements] Kristoff, J. and D. Wessels, "DNS Transport over TCP - Operational Requirements", draft-ietf-dnsop-dns-tcp-requirements-06 (work in progress), May 2020.
- [I-D.ietf-httpbis-bcp56bis] Nottingham, M., "Building Protocols with HTTP", draft-ietf-httpbis-bcp56bis-09 (work in progress), November 2019.
- [Internet.nl] Internet.nl, "Internet.nl Is Your Internet Up To Date?", 2019, <<https://internet.nl>>.
- [IP-Anonymization-in-Analytics] Google, "IP Anonymization in Analytics", 2019, <<https://support.google.com/analytics/answer/2763052?hl=en>>.
- [ipcipher1] Hubert, B., "On IP address encryption: security analysis with respect for privacy", 2017, <<https://medium.com/@bert.hubert/on-ip-address-encryption-security-analysis-with-respect-for-privacy-dabel1201b476>>.

- [ipcipher2] PowerDNS, "ipcipher", 2017, <<https://github.com/PowerDNS/ipcipher>>.
- [ipcrypt] veorq, "ipcrypt: IP-format-preserving encryption", 2015, <<https://github.com/veorq/ipcrypt>>.
- [ipcrypt-analysis] Aumasson, J., "Analysis of ipcrypt?", 2018, <<https://www.ietf.org/mail-archive/web/cfrg/current/msg09494.html>>.
- [ISC-Knowledge-database-on-cache-snooping] ISC Knowledge Database, "DNS Cache snooping - should I be concerned?", 2018, <<https://kb.isc.org/docs/aa-00482>>.
- [MAC-address-EDNS] DNS-OARC mailing list, "Embedding MAC address in DNS requests for selective filtering IDs", 2016, <<https://lists.dns-oarc.net/pipermail/dns-operations/2016-January/014143.html>>.
- [nginx] nginx.org, "NGINX", 2019, <<https://nginx.org/>>.
- [Passive-Observations-of-a-Large-DNS] de Vries, W., van Rijswijk-Deij, R., de Boer, P., and A. Pras, "Passive Observations of a Large DNS Service: 2.5 Years in the Life of Google", 2018, <http://tma.ifip.org/2018/wp-content/uploads/sites/3/2018/06/tma2018_paper30.pdf>.
- [pcap] tcpdump.org, "PCAP", 2016, <<http://www.tcpdump.org/>>.
- [Pitfalls-of-DNS-Encryption] Shulman, H., "Pretty Bad Privacy: Pitfalls of DNS Encryption", 2014, <<https://dl.acm.org/citation.cfm?id=2665959>>.
- [policy-comparison] dnsprivacy.org, "Comparison of policy and privacy statements 2019", 2019, <<https://dnsprivacy.org/wiki/display/DP/Comparison+of+policy+and+privacy+statements+2019>>.
- [PowerDNS-dnswasher] PowerDNS, "dnswasher", 2019, <<https://github.com/PowerDNS/pdns/blob/master/pdns/dnswasher.cc>>.

- [Ramaswamy-and-Wolf]
Ramaswamy, R. and T. Wolf, "High-Speed Prefix-Preserving IP Address Anonymization for Passive Measurement Systems", 2007,
<<http://www.ecs.umass.edu/ece/wolf/pubs/ton2007.pdf>>.
- [RFC4034] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "Resource Records for the DNS Security Extensions", RFC 4034, DOI 10.17487/RFC4034, March 2005,
<<https://www.rfc-editor.org/info/rfc4034>>.
- [RFC4035] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "Protocol Modifications for the DNS Security Extensions", RFC 4035, DOI 10.17487/RFC4035, March 2005,
<<https://www.rfc-editor.org/info/rfc4035>>.
- [RFC5077] Salowey, J., Zhou, H., Eronen, P., and H. Tschofenig, "Transport Layer Security (TLS) Session Resumption without Server-Side State", RFC 5077, DOI 10.17487/RFC5077, January 2008, <<https://www.rfc-editor.org/info/rfc5077>>.
- [RFC6235] Boschi, E. and B. Trammell, "IP Flow Anonymization Support", RFC 6235, DOI 10.17487/RFC6235, May 2011,
<<https://www.rfc-editor.org/info/rfc6235>>.
- [RFC6265] Barth, A., "HTTP State Management Mechanism", RFC 6265, DOI 10.17487/RFC6265, April 2011, <<https://www.rfc-editor.org/info/rfc6265>>.
- [RFC7626] Bortzmeyer, S., "DNS Privacy Considerations", RFC 7626, DOI 10.17487/RFC7626, August 2015, <<https://www.rfc-editor.org/info/rfc7626>>.
- [RFC7873] Eastlake 3rd, D. and M. Andrews, "Domain Name System (DNS) Cookies", RFC 7873, DOI 10.17487/RFC7873, May 2016,
<<https://www.rfc-editor.org/info/rfc7873>>.
- [RFC8027] Hardaker, W., Gudmundsson, O., and S. Krishnaswamy, "DNSSEC Roadblock Avoidance", BCP 207, RFC 8027, DOI 10.17487/RFC8027, November 2016, <<https://www.rfc-editor.org/info/rfc8027>>.
- [RFC8094] Reddy, T., Wing, D., and P. Patil, "DNS over Datagram Transport Layer Security (DTLS)", RFC 8094, DOI 10.17487/RFC8094, February 2017, <<https://www.rfc-editor.org/info/rfc8094>>.

- [RFC8404] Moriarty, K., Ed. and A. Morton, Ed., "Effects of Pervasive Encryption on Operators", RFC 8404, DOI 10.17487/RFC8404, July 2018, <<https://www.rfc-editor.org/info/rfc8404>>.
- [RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <<https://www.rfc-editor.org/info/rfc8446>>.
- [RFC8555] Barnes, R., Hoffman-Andrews, J., McCarney, D., and J. Kasten, "Automatic Certificate Management Environment (ACME)", RFC 8555, DOI 10.17487/RFC8555, March 2019, <<https://www.rfc-editor.org/info/rfc8555>>.
- [RFC8618] Dickinson, J., Hague, J., Dickinson, S., Manderson, T., and J. Bond, "Compacted-DNS (C-DNS): A Format for DNS Packet Capture", RFC 8618, DOI 10.17487/RFC8618, September 2019, <<https://www.rfc-editor.org/info/rfc8618>>.
- [SSL-Labs] SSL Labs, "SSL Server Test", 2019, <<https://www.ssllabs.com/ssltest/>>.
- [stunnel] ISC Knowledge Database, "DNS-over-TLS", 2018, <<https://kb.isc.org/article/AA-01386/0/DNS-over-TLS.html>>.
- [SURFnet-policy] SURFnet, "SURFnet Data Sharing Policy", 2016, <<https://surf.nl/datasharing>>.
- [TCPdpriv] Ipsilon Networks, Inc., "TCPdpriv", 2005, <<http://ita.ee.lbl.gov/html/contrib/tcpdpriv.html>>.
- [van-Dijkhuizen-et-al.] Van Dijkhuizen, N. and J. Van Der Ham, "A Survey of Network Traffic Anonymisation Techniques and Implementations", 2018, <<https://doi.org/10.1145/3182660>>.
- [Xu-et-al.] Fan, J., Xu, J., Ammar, M., and S. Moon, "Prefix-preserving IP address anonymization: measurement-based security evaluation and a new cryptography-based scheme", 2004, <<http://an.kaist.ac.kr/~sbmoon/paper/intl-journal/2004-cn-anon.pdf>>.

Appendix A. Documents

This section provides an overview of some DNS privacy-related documents, however, this is neither an exhaustive list nor a definitive statement on the characteristic of the document.

A.1. Potential increases in DNS privacy

These documents are limited in scope to communications between stub clients and recursive resolvers:

- o 'Specification for DNS over Transport Layer Security (TLS)' [RFC7858].
- o 'DNS over Datagram Transport Layer Security (DTLS)' [RFC8094]. Note that this document has the Category of Experimental.
- o 'DNS Queries over HTTPS (DoH)' [RFC8484].
- o 'Usage Profiles for DNS over TLS and DNS over DTLS' [RFC8310].
- o 'The EDNS(0) Padding Option' [RFC7830] and 'Padding Policy for EDNS(0)' [RFC8467].

These documents apply to recursive and authoritative DNS but are relevant when considering the operation of a recursive server:

- o 'DNS Query Name minimization to Improve Privacy' [RFC7816].

A.2. Potential decreases in DNS privacy

These documents relate to functionality that could provide increased tracking of user activity as a side effect:

- o 'Client Subnet in DNS Queries' [RFC7871].
- o 'Domain Name System (DNS) Cookies' [RFC7873]).
- o 'Transport Layer Security (TLS) Session Resumption without Server-Side State' [RFC5077] referred to here as simply TLS session resumption.
- o [RFC8446] Appendix C.4 describes Client Tracking Prevention in TLS 1.3
- o 'A DNS Packet Capture Format' [RFC8618].
- o Passive DNS [RFC8499].

- o Section 8 of [RFC8484] outlines the privacy considerations of DoH. Note that (while that document advises exposing the minimal set of data needed to achieve the desired feature set) depending on the specifics of a DoH implementation there may be increased identification and tracking compared to other DNS transports.

A.3. Related operational documents

- o 'DNS Transport over TCP - Implementation Requirements' [RFC7766].
- o 'Operational requirements for DNS over TCP' [I-D.ietf-dnsop-dns-tcp-requirements].
- o 'The edns-tcp-keepalive EDNS0 Option' [RFC7828].
- o 'DNS Stateful Operations' [RFC8490].

Appendix B. IP address techniques

The following table presents a high level comparison of various techniques employed or under development in 2019, and classifies them according to categorization of technique and other properties. Both the specific techniques and the categorisations are described in more detail in the following sections. The list of techniques includes the main techniques in current use, but does not claim to be comprehensive.

Categorization/Property	GA	d	TC	C	TS	i	B
Anonymization	X	X	X				X
Pseudoanonymization				X	X	X	
Format preserving	X	X	X	X	X	X	
Prefix preserving			X	X	X		
Replacement			X				
Filtering	X						
Generalization							X
Enumeration		X					
Reordering/Shuffling			X				
Random substitution			X				
Cryptographic permutation				X	X	X	
IPv6 issues					X		
CPU intensive				X			
Memory intensive			X				
Security concerns						X	

Table 1: Classification of techniques

Legend of techniques: GA = Google Analytics, d = dnswasher, TC = TCPdpriv, C = CryptoPAN, TS = TSA, i = ipcipher, B = Bloom filter

The choice of which method to use for a particular application will depend on the requirements of that application and consideration of the threat analysis of the particular situation.

For example, a common goal is that distributed packet captures must be in an existing data format such as PCAP [pcap] or C-DNS [RFC8618] that can be used as input to existing analysis tools. In that case, use of a format-preserving technique is essential. This, though, is not cost-free - several authors (e.g., [Brenker-and-Arnes] have observed that, as the entropy in an IPv4 address is limited, if an attacker can

- o ensure packets are captured by the target and
- o send forged traffic with arbitrary source and destination addresses to that target and
- o obtain a de-identified log of said traffic from that target

any format-preserving pseudonymization is vulnerable to an attack along the lines of a cryptographic chosen plaintext attack.

B.1. Categorization of techniques

Data minimization methods may be categorized by the processing used and the properties of their outputs. The following builds on the categorization employed in [RFC6235]:

- o Format-preserving. Normally when encrypting, the original data length and patterns in the data should be hidden from an attacker. Some applications of de-identification, such as network capture de-identification, require that the de-identified data is of the same form as the original data, to allow the data to be parsed in the same way as the original.
- o Prefix preservation. Values such as IP addresses and MAC addresses contain prefix information that can be valuable in analysis, e.g., manufacturer ID in MAC addresses, subnet in IP addresses. Prefix preservation ensures that prefixes are de-identified consistently; e.g., if two IP addresses are from the same subnet, a prefix preserving de-identification will ensure that their de-identified counterparts will also share a subnet. Prefix preservation may be fixed (i.e. based on a user selected prefix length identified in advance to be preserved) or general.

- o Replacement. A one-to-one replacement of a field to a new value of the same type, for example, using a regular expression.
- o Filtering. Removing or replacing data in a field. Field data can be overwritten, often with zeros, either partially (truncation or reverse truncation) or completely (black-marker anonymization).
- o Generalization. Data is replaced by more general data with reduced specificity. One example would be to replace all TCP/UDP port numbers with one of two fixed values indicating whether the original port was ephemeral (≥ 1024) or non-ephemeral (> 1024). Another example, precision degradation, reduces the accuracy of e.g., a numeric value or a timestamp.
- o Enumeration. With data from a well-ordered set, replace the first data item data using a random initial value and then allocate ordered values for subsequent data items. When used with timestamp data, this preserves ordering but loses precision and distance.
- o Reordering/shuffling. Preserving the original data, but rearranging its order, often in a random manner.
- o Random substitution. As replacement, but using randomly generated replacement values.
- o Cryptographic permutation. Using a permutation function, such as a hash function or cryptographic block cipher, to generate a replacement de-identified value.

B.2. Specific techniques

B.2.1. Google Analytics non-prefix filtering

Since May 2010, Google Analytics has provided a facility [IP-Anonymization-in-Analytics] that allows website owners to request that all their users IP addresses are anonymized within Google Analytics processing. This very basic anonymization simply sets to zero the least significant 8 bits of IPv4 addresses, and the least significant 80 bits of IPv6 addresses. The level of anonymization this produces is perhaps questionable. There are some analysis results [Geolocation-Impact-Assessment] which suggest that the impact of this on reducing the accuracy of determining the user's location from their IP address is less than might be hoped; the average discrepancy in identification of the user city for UK users is no more than 17%.

Anonymization: Format-preserving, Filtering (truncation).

B.2.2. dnswasher

Since 2006, PowerDNS have included a de-identification tool dnswasher [PowerDNS-dnswasher] with their PowerDNS product. This is a PCAP filter that performs a one-to-one mapping of end user IP addresses with an anonymized address. A table of user IP addresses and their de-identified counterparts is kept; the first IPv4 user addresses is translated to 0.0.0.1, the second to 0.0.0.2 and so on. The de-identified address therefore depends on the order that addresses arrive in the input, and running over a large amount of data the address translation tables can grow to a significant size.

Anonymization: Format-preserving, Enumeration.

B.2.3. Prefix-preserving map

Used in [TCPdpriv], this algorithm stores a set of original and anonymised IP address pairs. When a new IP address arrives, it is compared with previous addresses to determine the longest prefix match. The new address is anonymized by using the same prefix, with the remainder of the address anonymized with a random value. The use of a random value means that TCPdpriv is not deterministic; different anonymized values will be generated on each run. The need to store previous addresses means that TCPdpriv has significant and unbounded memory requirements, and because of the need to allocated anonymized addresses sequentially cannot be used in parallel processing.

Anonymization: Format-preserving, prefix preservation (general).

B.2.4. Cryptographic Prefix-Preserving Pseudonymization

Cryptographic prefix-preserving pseudonymization was originally proposed as an improvement to the prefix-preserving map implemented in TCPdpriv, described in [Xu-et-al.] and implemented in the [Crypto-PAN] tool. Crypto-PAN is now frequently used as an acronym for the algorithm. Initially it was described for IPv4 addresses only; extension for IPv6 addresses was proposed in [Harvan]. This uses a cryptographic algorithm rather than a random value, and thus pseudonymity is determined uniquely by the encryption key, and is deterministic. It requires a separate AES encryption for each output bit, so has a non-trivial calculation overhead. This can be mitigated to some extent (for IPv4, at least) by pre-calculating results for some number of prefix bits.

Pseudonymization: Format-preserving, prefix preservation (general).

B.2.5. Top-hash Subtree-replicated Anonymization

Proposed in [Ramaswamy-and-Wolf], Top-hash Subtree-replicated Anonymization (TSA) originated in response to the requirement for faster processing than Crypto-PAn. It used hashing for the most significant byte of an IPv4 address, and a pre-calculated binary tree structure for the remainder of the address. To save memory space, replication is used within the tree structure, reducing the size of the pre-calculated structures to a few Mb for IPv4 addresses. Address pseudonymization is done via hash and table lookup, and so requires minimal computation. However, due to the much increased address space for IPv6, TSA is not memory efficient for IPv6.

Pseudonymization: Format-preserving, prefix preservation (general).

B.2.6. ipcipher

A recently-released proposal from PowerDNS, ipcipher [ipcipher1] [ipcipher2] is a simple pseudonymization technique for IPv4 and IPv6 addresses. IPv6 addresses are encrypted directly with AES-128 using a key (which may be derived from a passphrase). IPv4 addresses are similarly encrypted, but using a recently proposed encryption [ipcrypt] suitable for 32bit block lengths. However, the author of ipcrypt has since indicated [ipcrypt-analysis] that it has low security, and further analysis has revealed it is vulnerable to attack.

Pseudonymization: Format-preserving, cryptographic permutation.

B.2.7. Bloom filters

van Rijswijk-Deij et al. have recently described work using Bloom filters [Bloom-filter] to categorize query traffic and record the traffic as the state of multiple filters. The goal of this work is to allow operators to identify so-called Indicators of Compromise (IOCs) originating from specific subnets without storing information about, or be able to monitor the DNS queries of an individual user. By using a Bloom filter, it is possible to determine with a high probability if, for example, a particular query was made, but the set of queries made cannot be recovered from the filter. Similarly, by mixing queries from a sufficient number of users in a single filter, it becomes practically impossible to determine if a particular user performed a particular query. Large numbers of queries can be tracked in a memory-efficient way. As filter status is stored, this approach cannot be used to regenerate traffic, and so cannot be used with tools used to process live traffic.

Anonymized: Generalization.

Appendix C. Current policy and privacy statements

A tabular comparison of policy and privacy statements from various DNS Privacy service operators based loosely on the proposed RPS structure can be found at [policy-comparison]. The analysis is based on the data available in December 2019.

We note that the existing set of policies vary widely in style, content and detail and it is not uncommon for the full text for a given operator to equate to more than 10 pages of moderate font sized A4 text. It is a non-trivial task today for a user to extract a meaningful overview of the different services on offer.

It is also noted that Mozilla have published a DoH resolver policy [DoH-resolver-policy], which describes the minimum set of policy requirements that a party must satisfy to be considered as a potential partner for Mozilla's Trusted Recursive Resolver (TRR) program.

Appendix D. Example RPS

The following example RPS is very loosely based on some elements of published privacy statements for some public resolvers, with additional fields populated to illustrate the what the full contents of an RPS might look like. This should not be interpreted as

- o having been reviewed or approved by any operator in any way
- o having any legal standing or validity at all
- o being complete or exhaustive

This is a purely hypothetical example of an RPS to outline example contents - in this case for a public resolver operator providing a basic DNS Privacy service via one IP address and one DoH URI with security based filtering. It does aim to meet minimal compliance as specified in Section 5.

D.1. Policy

1. Treatment of IP addresses. Many nations classify IP addresses as personal data, and we take a conservative approach in treating IP addresses as personal data in all jurisdictions in which our systems reside.
2. Data collection and sharing.

1. IP addresses. Our normal course of data management does not have any IP address information or other personal data logged to disk or transmitted out of the location in which the query was received. We may aggregate certain counters to larger network block levels for statistical collection purposes, but those counters do not maintain specific IP address data nor is the format or model of data stored capable of being reverse-engineered to ascertain what specific IP addresses made what queries.
2. Data collected in logs. We do keep some generalized location information (at the city/metropolitan area level) so that we can conduct debugging and analyze abuse phenomena. We also use the collected information for the creation and sharing of telemetry (timestamp, geolocation, number of hits, first seen, last seen) for contributors, public publishing of general statistics of system use (protections, threat types, counts, etc.) When you use our DNS Services, here is the full list of items that are included in our logs:

- + Request domain name, e.g., example.net
- + Record type of requested domain, e.g., A, AAAA, NS, MX, TXT, etc.
- + Transport protocol on which the request arrived, i.e. UDP, TCP, DoT, DoH
- + Origin IP general geolocation information: i.e. geocode, region ID, city ID, and metro code
- + IP protocol version - IPv4 or IPv6
- + Response code sent, e.g., SUCCESS, SERVFAIL, NXDOMAIN, etc.
- + Absolute arrival time using a precision in ms
- + Name of the specific instance that processed this request
- + IP address of the specific instance to which this request was addressed (no relation to the requestor's IP address)

We may keep the following data as summary information, including all the above EXCEPT for data about the DNS record requested:

- + Currently-advertised BGP-summarized IP prefix/netmask of apparent client origin
- + Autonomous system number (BGP ASN) of apparent client origin

All the above data may be kept in full or partial form in permanent archives.

3. Sharing of data. Except as described in this document, we do not intentionally share, sell, or rent individual personal information associated with the requestor (i.e. source IP address or any other information that can positively identify the client using our infrastructure) with anyone without your consent. We generate and share high level anonymized aggregate statistics including threat metrics on threat type, geolocation, and if available, sector, as well as other vertical metrics including performance metrics on our DNS Services (i.e. number of threats blocked, infrastructure uptime) when available with our threat intelligence (TI) partners, academic researchers, or the public. Our DNS Services share anonymized data on specific domains queried (records such as domain, timestamp, geolocation, number of hits, first seen, last seen) with our threat intelligence partners. Our DNS Services also builds, stores, and may share certain DNS data streams which store high level information about domain resolved, query types, result codes, and timestamp. These streams do not contain IP address information of requestor and cannot be correlated to IP address or other personal data. We do not and never will share any of its data with marketers, nor will it use this data for demographic analysis.
3. Exceptions. There are exceptions to this storage model: In the event of actions or observed behaviors which we deem malicious or anomalous, we may utilize more detailed logging to collect more specific IP address data in the process of normal network defence and mitigation. This collection and transmission off-site will be limited to IP addresses that we determine are involved in the event.
4. Associated entities. Details of our Threat Intelligence partners can be found at our website page (insert link).
5. Correlation of Data. We do not correlate or combine information from our logs with any personal information that you have provided us for other services, or with your specific IP address.

6. Result filtering.

1. Filtering. We utilise cyber threat intelligence about malicious domains from a variety of public and private sources and blocks access to those malicious domains when your system attempts to contact them. An NXDOMAIN is returned for blocked sites.
1. Censorship. We will not provide a censoring component and will limit our actions solely to the blocking of malicious domains around phishing, malware, and exploit kit domains.
2. Accidental blocking. We implement allowlisting algorithms to make sure legitimate domains are not blocked by accident. However, in the rare case of blocking a legitimate domain, we work with the users to quickly allowlist that domain. Please use our support form ([insert link](#)) if you believe we are blocking a domain in error.

D.2. Practice

1. Deviations from Policy. None in place since (insert date).
2. Client facing capabilities.
 1. We offer UDP and TCP DNS on port 53 on (insert IP address)
 2. We offer DNS over TLS as specified in RFC7858 on (insert IP address). It is available on port 853 and port 443. We also implement RFC7766.
 1. The DoT authentication domain name used is (insert domain name).
 2. We do not publish SPKI pin sets.
 3. We offer DNS over HTTPS as specified in RFC8484 on (insert URI template).
 4. Both services offer TLS 1.2 and TLS 1.3.
 5. Both services pad DNS responses according to RFC8467.
 6. Both services provide DNSSEC validation.
3. Upstream capabilities.

1. Our servers implement QNAME minimization.
2. Our servers do not send ECS upstream.
4. Support. Support information for this service is available at (insert link).
5. Data Processing. We operate as the legal entity (insert entity) registered in (insert country); as such we operate under (insert country/region) law. Our separate statement regarding the specifics of our data processing policy, practice, and agreements can be found here (insert link).

Authors' Addresses

Sara Dickinson
Sinodun IT
Magdalen Centre
Oxford Science Park
Oxford OX4 4GA
United Kingdom

Email: sara@sinodun.com

Benno J. Overeinder
NLnet Labs
Science Park 400
Amsterdam 1098 XH
The Netherlands

Email: benno@nlnetLabs.nl

Roland M. van Rijswijk-Deij
NLnet Labs
Science Park 400
Amsterdam 1098 XH
The Netherlands

Email: roland@nlnetLabs.nl

Allison Mankin
Salesforce

Email: allison.mankin@gmail.com