Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: May 4, 2020

E. Kinnear
T. Pauly
C. Wood
Apple Inc.
P. McManus
Fastly

November 01, 2019

Adaptive DNS: Improving Privacy of Name Resolution
draft-pauly-dprive-adaptive-dns-privacy-01

Abstract

This document defines an architecture that allows clients to dynamically discover designated resolvers that offer encrypted DNS services, and use them in an adaptive way that improves privacy while co-existing with locally provisioned resolvers. These resolvers can be used directly when looking up names for which they are designated. These resolvers also provide the ability to proxy encrypted queries, thus hiding the identity of the client requesting resolution.

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 May 4, 2020.

Copyright Notice

Table of Contents

1. Introduction

When clients need to resolve names into addresses in order to establish networking connections, they traditionally use by default the DNS resolver that is provisioned by the local network along with their IP address [RFC2132] [RFC8106]. Alternatively, they can use a resolver indicated by a tunneling service such as a VPN.

However, privacy-sensitive clients might prefer to use an encrypted DNS service other than the one locally provisioned in order to prevent interception, profiling, or modification by entities other than the operator of the name service for the name being resolved. Protocols that can improve the transport security of a client when using DNS or creating TLS connections include DNS-over-TLS [RFC7858], DNS-over-HTTPS [RFC8484], and encrypted Server Name Indication (ESNI) [I-D.ietf-tls-esni].

There are several concerns around a client using such privacy-enhancing mechanisms for generic system traffic. A remote service that provides encrypted DNS may not provide correct answers for locally available resources, or private resources (such as domains only accessible over a private network). Remote services may also be untrusted from a privacy perspective: while encryption will prevent on-path observers from seeing hostnames, client systems need to trust the encrypted DNS service to not store or misuse queries made to it. Further, extensive use of cloud based recursive resolvers obscures the network location of the client which may degrade the performance of the returned server due to lack of proximity at the benefit of improved privacy.

Client systems are left with choosing between one of the following stances:

1. Send all application DNS queries to a particular encrypted DNS service, which requires establishing user trust of the service. This can lead to resolution failures for local or private enterprise domains absent heuristics or other workarounds for detecting managed networks.

2. Allow the user or another entity to configure local policy for which domains to send to local, private, or encrypted resolvers. This provides more granularity at the cost of increasing user burden.

3. Only use locally-provisioned resolvers, and opportunistically use encrypted DNS to these resolvers when possible. (Clients may

learn of encrypted transport support by actively probing such resolvers.) This provides marginal benefit over not using encrypted DNS at all, especially if clients have no means of authenticating or trusting local resolvers.

This document defines an architecture that allows clients to improve the privacy of their DNS queries without requiring user intervention, and allowing coexistence with local, private, and enterprise resolvers.

This architecture is composed of several mechanisms:

- A DNS record that indicates a designated DoH server associated with a name (Section 3.1);
- an extension to DoH that allows client IP addresses to be disassociated from queries via proxying ([I-D.pauly-dprive-oblivious-doh]);
- a DoH server that responds to queries directly and supports proxying (Section 4);
- and client behavior rules for how to resolve names using a combination of designated DoH resolvers, proxied queries, and local resolvers (Section 3).

1.1. Specification of Requirements

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.

2. Terminology

This document defines the following terms:

Adaptive DNS: Adaptive DNS is a technique to provide an encrypted transport for DNS queries that can be sent directly to a Designated DoH Server, to use Oblivious DoH to hide the client IP address, or to use Direct Resolvers when required or appropriate.

Designated DoH Server: A DNS resolver that provides connectivity over HTTPS (DoH) that is designated as a responsible resolver for a given domain or zone.

Direct Resolver: A DNS resolver using any transport that is provisioned directly by a local router or a VPN.

Exclusive Direct Resolver: A Direct Resolver that requires the client to use it exclusively for a given set of domains, such as private domains managed by a VPN. This status is governed by local system policy.

Oblivious DoH: A technique that uses multiple DoH servers to proxy queries in a way that disassociates the client's IP address from query content.

Oblivious Proxy: A resolution server that proxies encrypted client DNS queries to another resolution server that will be able to decrypt the query (the Oblivious Target).

Oblivious Target: A resolution server that receives encrypted client DNS queries via an Oblivious Proxy.

Privacy-Sensitive Connections: Connections made by clients that are explicitly Privacy-Sensitive are treated differently from connections made for generic system behavior, such as non-user-initiated maintenance connections. This distinction is only relevant on the client, and does not get communicated to other network entities. Certain applications, such as browsers, can choose to treat all connections as privacy-sensitive.

Web PvD: A Web Provisioning Domain, or Web PvD, represents the configuration of resolvers, proxies, and other information that a server deployment makes available to clients. See Section 4.3.

3. Client Behavior

Adaptive DNS allows client systems and applications to improve the privacy of their DNS queries and connections, both by requiring confidentiality via encryption, and by limiting the ability to correlate client IP addresses with query contents. Specifically, the goal for client queries is to achieve the following properties:

1. No party other than the client and server can learn or control the names being queried by the client or the answers being returned by the server.

2. Only a designated DNS resolver associated with the deployment that is also hosting content will be able to read both the client IP address and queried names for Privacy-Sensitive Connections. For example, a resolver owned and operated by the same provider that hosts "example.com" would be able to link queries for

"example.com" to specific clients (by their IP address), since the server ultimately has this capability once clients subsequently establish secure (e.g., TLS) connections to an address to which "example.com" resolves.

3. Clients will be able to comply with policies required by VPNs and local networks that are authoritative for private domains.

An algorithm for determining how to resolve a given name in a manner that satisfies these properties is described in Section 3.3. Note that this algorithm does not guarantee that responses that are not signed with DNSSEC are valid, and clients that establish connections to unsigned addresses may still expose their local IP addresses to attackers that control their terminal resolver even if hidden during resolution.

3.1. Discovering Designated DoH Servers

All direct (non-oblivious) queries for names in privacy-sensitive connections MUST be sent to a server that both provides encryption and is designated for the domain.

Clients dynamically build and maintain a set of known Designated DoH Servers. The information that is associated with each server is:

- The URI Template of the DoH server [RFC8484]
- The public HPKE [I-D.irtf-cfrg-hpke] key of the DoH server used for proxied oblivious queries [I-D.pauly-dprive-oblivious-doh]
- A list of domains for which the DoH server is designated

This information can be retrieved from several different sources. The primary source for discovering Designated DoH Server configurations is from properties stored in a SVCB (or a SVCB-conformant type like HTTPSSVC) DNS Record [I-D.nygren-dnsop-svcb-httpssvc]. This record provides the URI Template and the public Oblivious DoH key of a DoH server that is designated for a specific domain. A specific domain may have more than one such record.

In order to designate a DoH server for a domain, a SVCB record can contain the "dohuri" (Section 10). The value stored in the parameter is a URI, which is the DoH URI template [RFC8484].

The public key of the DoH server is sent as the "odohkey" [I-D.pauly-dprive-oblivious-doh].

The following example shows a record containing a DoH URI, as returned by a query for the HTTPSSVC variant of the SVCB record type on "example.com".

```
example.com.      7200  IN HTTPSSVC 0 svc.example.net.
svc.example.net.  7200  IN HTTPSSVC 2 svc1.example.net. (
                                      dohuri=https://doh.example.net/dns-query
                                      odohkey="..." )
```

Clients MUST ignore any DoH server URI that was not retrieved from a DNSSEC-signed record that was validated by the client [RFC4033].

Whenever a client resolves a name for which it does not already have a Designated DoH Server, it SHOULD try to determine the Designated DoH Server by sending a query for the an SVCB record for the name. If there is no DoH server designated for the name or zone, signaled either by an NXDOMAIN answer or a SVCB record that does not contain a DoH URI, the client SHOULD suppress queries for the SVCB record for a given name until the time-to-live of the answer expires.

In order to bootstrap discovery of Designated DoH Servers, client systems SHOULD have some saved list of at least two names that they use consistently to perform SVCB record queries on the Direct Resolvers configured by the local network. Since these queries are likely not private, they SHOULD NOT be associated with user action or contain user-identifying content. Rather, the expectation is that all client systems of the same version and configuration would issue the same bootstrapping queries when joining a network for the first time when the list of Designated DoH Servers is empty.

3.1.1. Whitelisting Designated DoH Servers

Prior to using a Designated DoH Server for direct name queries on privacy-sensitive connections, clients MUST whitelist the server.

The requirements for whitelisting are:

- Support for acting as an Oblivious Proxy. Each Designated DoH Server is expected to support acting as a proxy for Oblivious DoH. A client MUST issue at least one query that is proxied through the server before sending direct queries to the server.

- Support for acting as an Oblivious Target. Each Designated DoH Server is expected to support acting as a target for Oblivious DoH. A client MUST issue at least one query that is targeted at the server through a proxy before sending direct queries to the server.

Designated DoH Servers are expected to act both as Oblivious Proxies and as Oblivious Targets to ensure that clients have sufficient options for preserving privacy using Oblivious DoH. Oblivious Targets are expected to act as Oblivious Proxies to ensure that no Oblivious DoH server can act as only a target (thus being able to see patterns in name resolution, which might have value to a resolver) and require other servers to take on a disproportionate load of proxying.

Clients MAY further choose to restrict the whitelist by other local policy. For example, a client system can have a list of trusted resolver configurations, and it can limit the whitelist of Designated DoH Servers to configurations that match this list. Alternatively, a client system can check a server against a list of audited and approved DoH Servers that have properties that the client approves.

Clients SHOULD NOT whitelist authority mappings for effective top-level domains (eTLDs), such as ".com".

If a client detects at any point after whitelisting a DoH server that the server no longer meets the criteria for whitelisting, such as consistently failing to proxy or receive Oblivious DoH queries, the client SHOULD remove the DoH server from its whitelist.

3.1.2. Accessing Extended Information

When a Designated DoH Server is discovered, clients SHOULD also check to see if this server provides an extended configuration in the form of a Web PvD (Section 4.3). To do this, the client performs a GET request to the DoH URI, indicating that it accepts a media type of "application/pvd+json" [I-D.ietf-intarea-provisioning-domains]. When requesting the PvD information, the query and fragment components of the requested path are left empty. Note that this is different from a GET request for the "application/dns-message" type, in which the query variable "dns" contains an encoded version of a DNS message.

In response, the server will return the JSON content for the PvD, if present. The content-type MUST be "application/pvd+json".

The following exchange shows an example of a client retrieving a Web PvD configuration for a DoH server with the URI Template "https://dnsserver.example.net/dns-query".

The client sends:

```
:method = GET
:scheme = https
:authority = dnsserver.example.net
:path = /dns-query
accept = application/pvd+json

And the server replies:

:status = 200
content-type = application/pvd+json
content-length = 175
cache-control = max-age=86400

<JSON content of the Web PvD>
```

If the server does not support retrieving any extended PvD information, it MUST reply with HTTP status code 415 (Unsupported Media Type, [RFC7231]).

If the retrieved JSON contains a "dnsZones" array [I-D.ietf-intarea-provisioning-domains], the client SHOULD perform an SVCB record lookup of each of the listed zones on the DoH server and validate that the DoH server is a designated server for the domain; and if it is, add the domain to the local configuration.

3.2. Discovering Local Resolvers

If the local network provides configuration with an Explicit Provisioning Domain (PvD), as defined by [I-D.ietf-intarea-provisioning-domains], clients can learn about domains for which the local network's resolver is authoritative.

If an RA provided by the router on the network defines an Explicit PvD that has additional information, and this additional information JSON dictionary contains the key "dohTemplate" (Section 10), then the client SHOULD add this DoH server to its list of known DoH configurations. The domains that the DoH server claims authority for are listed in the "dnsZones" key. Clients MUST use an SVCB record from the locally-provisioned DoH server and validate the answer with DNSSEC [RFC4033] before creating a mapping from the domain to the server. Once this has been validated, clients can use this server for resolution as described in step 2 of Section 3.3.

See Section 6 for local deployment considerations.

3.3. Hostname Resolution Algorithm

When establishing a secure connection to a certain hostname, clients need to first determine which resolver configuration ought to be used for DNS resolution.

Several of the steps outlined in this algorithm take into account the success or failure of name resolution. Failure can be indicated either by a DNS response, such as SERVFAIL or NXDOMAIN, or by a connection-level failure, such as a TCP reset, TLS handshake failure, or an HTTP response error status. In effect, any unsuccessful attempt to resolve a name can cause the client to try another resolver if permitted by the algorithm. This is particularly useful for cases in which a name may not be resolvable over public DNS but has a valid answer only on the local network.

Given a specific hostname, the order of preference for which resolver to use SHOULD be:

1. An Exclusive Direct Resolver, such as a resolver provisioned by a VPN, with domain rules that include the hostname being resolved. If the resolution fails, the connection will fail. See Section 3.2 and Section 6.

2. A Direct Resolver, such as a local router, with domain rules that are known to be authoritative for the domain containing the hostname. If the resolution fails, the connection will try the next resolver configuration based on this list.

3. The most specific Designated DoH Server that has been whitelisted (Section 3.1.1) for the domain containing the hostname, i.e., the designated DoH server which is associated with the longest matching suffix of the hostname. For example, given two Designated DoH Servers, one for "foo.example.com" and another "example.com", clients connecting to "bar.foo.example.com" should use the former. Note that the matching MUST be performed on entire labels. That is, if "example.com" has a designated DoH server, it can be used for "foo.example.com", but not for "badexample.com". If the resolution fails, the connection can either use an Oblivious DoH resolver (Step 4) or the default resolver (Step 5). Privacy-Sensitive Connections SHOULD NOT skip Step 4. Other connections MAY skip Step 4, based on system policy.

4. Oblivious DoH queries using multiple DoH Servers ([I-D.pauly-dprive-oblivious-doh]). If this resolution fails, Privacy-Sensitive Connections will fail. All other connections will use the last resort, the default Direct Resolvers.

5. The default Direct Resolver, generally the resolver provisioned by the local router, is used as the last resort for any connection that is not explicitly Privacy-Sensitive [RFC2132] [RFC8106].

If the system allows the user to specify a preferred encrypted resolver, such as allowing the user to manually configure a DoH server URI to use by default, the use of this resolver SHOULD come between steps 2 and 3. This ensures that VPN-managed and locally-accessible names remain accessible while all other names are resolved using the user preference.

Resolution on behalf of system traffic, such as interactions required to detect and access a Captive Network Portal, require the use of the default Direct Resolver. System traffic SHOULD have an exception to this algorithm, and only use Steps 2 and 5 (those that use a resolver provisioned by the local network). Further deployment considerations for captive networks and walled-garden networks can be found in Section 6.2.3.

3.4. Oblivious Resolution

For all privacy-sensitive connection queries for names that do not correspond to a Designated DoH Server, the client SHOULD use Oblivious DoH to help conceal its IP address from eavesdroppers and untrusted resolvers.

Disassociation of client IPs from query content is achieved by using Oblivious DoH [I-D.pauly-dprive-oblivious-doh]. This extension to DoH allows a client to encrypt a query with a target DoH server's public key, and proxy the query through another server. The query is packaged with a unique client-defined symmetric key that is used to sign the DNS answer, which is sent back to the client via the proxy.

All DoH Servers that are used as Designated DoH Servers by the client MUST support being both an Oblivious Proxy and an Oblivious Target, as described in the server requirements (Section 4).

Since each Designated DoH Server can act as one of two roles in an proxied exchange, there are (N) * (N - 1) / 2 possible pairs of servers, where N is the number of whitelisted servers. While clients SHOULD use a variety of server pairs in rotation to decrease the ability for any given server to track client queries, it is not expected that all possible combinations will be used. Some combinations will be able to handle more load than others, and some will have better latency properties than others. To optimize performance, clients SHOULD maintain statistics to track the performance characteristics and success rates of particular pairs.

Clients that are performing Oblivious DoH resolution SHOULD fall back to another pair of servers if a first query times out, with a locally-determined limit for the number of fallback attempts that will be performed.

3.5. Handling Network Changes

Whenever a client joins a new network, it SHOULD wait to receive local configuration for resolvers before using any Designated DoH servers. The local network might be authoritative for some names, or might require filtering.

Once the local configuration of the new network has been received, the client MAY use Designated DoH configuration that it discovered when associated with another network. These configurations can still be considered valid since they came from DNSSEC-signed records. However, it is possible that different resolver IP addresses would be returned when looking up the designated server on the new network, which can provide a more optimal route through the Internet, so clients SHOULD perform new queries to refresh their mappings by making queries on connection on this new interface.

4. Server Requirements

Any server deployment that provides a set of services within one or more domains, such as a CDN, can run a server node that allows clients to run Adaptive DNS. A new server node can be added at any time, and can be used once it is advertised to clients and can be validated and whitelisted. The system overall is intended to scale and provide improved performance as more nodes become available.

The basic requirements to participate as a server node in this architecture are described below.

4.1. Provide a DoH Server

Each server node is primarily defined by a DoH server [RFC8484] that is designated for a set of domains, and also provides Oblivious DoH functionality. As such, the DoH servers MUST be able to act as recursive resolvers that accept queries for records and domains beyond those for which the servers are specifically designated.

4.1.1. Oblivious DoH Proxy

The DoH servers MUST be able to act as Oblivious Proxies. In this function, they will proxy encrypted queries and answers between clients and Oblivious Target DoH servers.

4.1.2. Oblivious DoH Target

The DoH servers MUST be able to act as Oblivious Targets. In this function, they will accept encrypted proxied queries from clients via Oblivious Proxy DoH servers, and provide encrypted answers using client keys.

4.1.3. Keying Material

In order to support acting as an Oblivious Target, a DoH server needs to provide a public HPKE [I-D.irtf-cfrg-hpke] key that can be used to encrypt client queries. This key is advertised in the SVCB record [I-D.pauly-dprive-oblivious-doh].

DoH servers also SHOULD provide an ESNI [I-D.ietf-tls-esni] key to encrypt the Server Name Indication field in TLS handshakes to the DoH server.

4.2. Advertise the DoH Server

The primary mechanism for advertising a Designated DoH Server is a SVCB DNS record (Section 3.1). This record MUST contain both the URI Template of the DoH Server as well as the Oblivious DoH Public Key. It MAY contain the ESNI key [I-D.ietf-tls-esni].

Servers MUST ensure that any SVCB records are signed with DNSSEC [RFC4033].

4.3. Provide Extended Configuration as a Web PvD

Beyond providing basic DoH server functionality, server nodes SHOULD provide a mechanism that allows clients to look up properties and configuration for the server deployment. Amongst other information, this configuration can optionally contain a list of some popular domains for which this server is designated. Clients can use this list to optimize lookups for common names.

This set of extended configuration information is referred to as a Web Provisioning Domain, or a Web PvD. Provisioning Domains are sets of consistent information that clients can use to access networks, including rules for resolution and proxying. Generally, these PvDs are provisioned directly, such as by a local router or a VPN. [I-D.ietf-intarea-provisioning-domains] defines an extensible configuration dictionary that can be used to add information to local PvD configurations. Web PvDs share the same JSON configuration format, and share the registry of keys defined as "Additional Information PvD Keys".

If present, the PvD JSON configuration MUST be made available to clients that request the "application/pvd+json" media type in a GET request to the DoH server's URI [I-D.ietf-intarea-provisioning-domains]. Clients MUST include this media type as an Accept header in their GET requests, and servers MUST mark this media type as their Content-Type header in responses. If the PvD JSON format is not supported, the server MUST reply with HTTP status code 415 [RFC7231].

The "identifier" key in the JSON configuration SHOULD be the hostname of the DoH Server itself.

For Web PvDs, the "prefixes" key within the JSON configuration SHOULD contain an empty array.

The key "dnsZones", which contains an array of domains as strings [I-D.ietf-intarea-provisioning-domains], indicates the zones that belong to the PvD. Any zone that is listed in this array for a Web PvD MUST have a corresponding SVCB record that defines the DoH server as designated for the zone. Servers SHOULD include in this array any names that are considered default or well-known for the deployment, but is not required or expected to list all zones or domains for which it is designated. The trade-off here is that zones that are listed can be fetched and validated automatically by clients, thus removing a bootstrapping step in discovering mappings from domains to Designated DoH Servers.

Clients that retrieve the Web PvD JSON dictionary SHOULD perform an SVCB record query for each of the entries in the "dnsZones" array in order to populate the mappings of domains. These MAY be performed in an oblivious fashion, but MAY also be queried directly on the DoH server (since the information is not user-specific, but in response to generic server-driven content). Servers can choose to preemptively transfer the relevant SVCB records if the PvD information retrieval is done with an HTTP version that supports PUSH semantics. This allows the server to avoid a round trip in zone validation even before the client has started requested SVCB records. Once the client requests an SVCB record for one of the names included in the "dnsZones" array, the server can also include the SVCB records for the other names in the array in the Additionals section of the DNS response.

This document also registers one new key in the Additional Information PvD Keys registry, to identify the URI Template for the DoH server Section 10. When included in Web PvDs, this URI MUST match the template in the SVCB DNS Record.

Beyond providing resolution configuration, the Web PvD configuration can be extended to offer information about proxies and other services offered by the server deployment. Such keys are not defined in this document.

5. Server Deployment Considerations

When servers designate DoH servers for their names, the specific deployment model can impact the effective privacy and performance characteristics.

5.1. Single Content Provider

If a name always resolves to server IP addresses that are hosted by a single content provider, the name ought to designate a single DoH server. This DoH server will be most optimal when it is designated by many or all names that are hosted by the same content provider. This ensures that clients can increase connection reuse to reduce latency in connection setup.

A DoH server that corresponds to the content provider that hosts content has an opportunity to tune the responses provided to a client based on the location inferred by the client IP address.

5.2. Multiple Content Providers

Some hostnames may resolve to server IP addresses that are hosted by multiple content providers. In such scenarios, the deployment may want to be able to control the percentage of traffic that flows to each content provider.

In these scenarios, there can either be:

- multiple designated DoH servers that are advertised via SVCB DNS Records; or,
- a single designated DoH server that can be referenced by one or more SVCB DNS Records, operated by a party that is aware of both content providers and can manage splitting the traffic.

If a server deployment wants to easily control the split of traffic between different content providers, it ought to use the latter model of using a single designated DoH server that can better control which IP addresses are provided to clients. Otherwise, if a client is aware of multiple DoH servers, it might use a single resolver exclusively, which may lead to inconsistent behavior between clients that choose different resolvers.

5.3. Avoid Narrow Deployments

Using designated DoH servers can improve the privacy of name resolution whenever a DoH server is designated by many different names within one or more domains. This limits the amount of information leaked to an attacker observing traffic between a client and a DoH server: the attacker only learns that the client might be resolving one of the many names for which the server is designated (or might be performing an Oblivious query).

However, if a deployment designates a given DoH server for only one name, or a very small set of names, then it becomes easier for an attacker to infer that a specific name is being accessed by a client. For this reason, deployments are encouraged to avoid deploying a DoH server that is only designated by a small number of names. Clients can also choose to only whitelist DoH servers that are associated with many names.

Beyond the benefits to privacy, having a larger number of names designate a given DoH server improves the opportunity for DoH connection reuse, which can improve the performance of name resolutions.

6. Local Resolver Deployment Considerations

A key goal of Adaptive DNS is that clients will be able to use Designated DoH Servers to improve the privacy of queries, without entirely bypassing local network authority and policy. For example, if a client is attached to an enterprise Wi-Fi network that provides access and resolution for private names not generally accessible on the Internet, such names will only be usable when a local resolver is used.

In order to achieve this, a local network can advertise itself as authoritative for a domain, allowing it to be used prior to external servers in the client resolution algorithm Section 3.3.

6.1. Designating Local DoH Servers

If a local network wants to have clients send queries for a set of private domains to its own resolver, it needs to define an explicit provisioning domain [I-D.ietf-intarea-provisioning-domains]. The PvD RA option SHOULD set the H-flag to indicate that Additional Information is available. This Additional Information JSON object SHOULD include both the "dohTemplate" and "dnsZones" keys to define the local DoH server and the domains over which it claims authority.

In order to validate that a local resolver is designated for a given zone, the client SHOULD issue a SVCB record query for the names specified in the PvD information, using the DoH server specified in the PvD information. If there is no SVCB record for a name that points to the DoH server that can be validated using DNSSEC, the client SHOULD NOT automatically create a designation from the domain name to DoH server. See specific use cases in Section 6.2 for cases in which a local resolver may still be used.

Although local Designated DoH Servers MAY support proxying Oblivious DoH queries, a client SHOULD NOT select one of these servers as an Oblivious Proxy. Doing so might reveal the client's location to the Target based on the address of the proxy, which could contribute to deanonymizing the client. Clients can make an exception to this behavior if the DoH server designated by the local network is known to be a non-local service, such as when a local network configures a centralized public resolver to handle its DNS operations.

6.2. Local Use Cases

The various use cases for selecting locally-provisioned resolvers require different approaches for deployment and client resolution. The following list is not exhaustive, but provides guidance on how these scenarios can be achieved using the Adaptive DNS algorithm.

6.2.1. Accessing Local-Only Resolvable Content

Some names are not resolvable using generic DNS resolvers, but require using a DNS server that can resolve private names. This is common in enterprise scenarios, in which an enterprise can have a set of private names that it allows to be resolved when connected to a VPN or an enterprise-managed Wi-Fi network. In this case, clients that do not use the locally-provisioned resolver will fail to resolve private names.

In these scenarios, the local network SHOULD designate a local DoH server for the domains that are locally resolvable. For example, an enterprise that owns "private.example.org" would advertise "private.example.org" in its PvD information along with a DoH URI template. Clients could then use that locally-configured resolver with names under "private.example.org" according to the rules in Section 6.1.

In general, clients SHOULD only create designated DoH server associations when they can validate a SVCB record using DNSSEC. However, some deployments of private names might not want to sign all private names within a zone. There are thus a few possible deployment models:

- "private.example.org" does have a DNSSEC-signed SVCB record that points to the local DoH server. The client requests the SVCB record for "private.example.org" using the local DoH server that is specified in the PvD information, and from that point on uses the local DoH server for names under "private.example.org".

- Instead of signing "private.example.org", the deployment provides a DNSSEC-signed SVCB record for "example.org", thus steering all resolution under "example.org" to the local resolver.

- No DNSSEC-signed SVCB record designates the local server. In this case, clients have a hint that the local network can serve names under "private.example.org", but do not have a way to validate the designation. Clients can in this case try to resolve names using external servers (such as via Oblivious DoH), and then MAY fall back to using locally-provisioned resolvers if the names do not resolve externally. This approach has the risk of exposing private names to public resolvers, which can be undesirable for certain enterprise deployments. Alternatively, if the client trusts the local network based on specific policy configured on the client, it can choose to resolve these names locally first. Note that this approach risks exposing names to a potentially malicious network that is masquerading as an authority for private names if the network cannot be validated in some other manner.

Deployments SHOULD use the one of the first two approaches (signing their records) whenever possible; the case of providing unsigned names is only described as a possibility for handling legacy enterprise deployments. Clients SHOULD choose to ignore any locally designated names that are not signed unless there is a specific policy configuration on the client.

6.2.2. Accessing Locally Optimized Content

Other names may be resolvable both publicly and on the local resolver, but have more optimized servers that are accessible only via the local network. For example, a Wi-Fi provider may provide access to a cache of video content that provides lower latency than publicly-accessible caches.

Names that are hosted locally in this way SHOULD use a designation with a DNSSEC-signed SVCB record for the name. If a client discovers that a local resolver is designated for a given name, the client SHOULD prefer using connections to this locally-hosted content rather than names resolved externally.

Note that having a DNSSEC-signed designation to the local resolver provides a clear indication that the entity that manages a given name has an explicit relationship with the local network provider.

6.2.3. Walled-Garden and Captive Network Deployments

Some networks do not provide any access to the general Internet, but host local content that clients can access. For example, a network on an airplane can give access to flight information and in-flight media, but will not allow access to any external hosts or DNS servers. These networks are often described as "walled-gardens".

Captive networks [I-D.ietf-capport-architecture] are similar in that they block access to external hosts, although they can provide generic access after some time.

If a walled-garden or captive network defines a PvD with additional information, it can define zones for names that it hosts, such as "airplane.example.com". It can also provide a locally-hosted encrypted DNS server.

However, if such a network does not support explicitly advertising local names, clients that try to establish connections to DoH servers will experience connection failures. In these cases, system traffic that is used for connecting to captive portals SHOULD use local resolvers. In addition, clients MAY choose to fall back to using direct resolution without any encryption if they determine that all connectivity is blocked otherwise. Note that this comes with a risk of a network blocking connections in order to induce this fallback behavior, so clients might want to inform users about this possible attack where appropriate, or prefer to not fall back if there is a concern about leaking user data.

6.2.4. Network-Based Filtering

Some networks currently rely on manipulating DNS name resolution in order to apply content filtering rules to clients associated with the network. Using encrypted DNS resolvers that are not participating in this filtering can bypass such enforcement. However, simply blocking connections for filtering is indistinguishable from a malicious attack from a client's perspective.

In order to indicate the presence of filtering requirements, a network deployment can add the "requireDNSFiltering" and "dnsFilteredZones" keys to its PvD information. The "dnsFilteredZones" entry can contain an array of strings, each of which is a domain name that the network requires clients to resolve using the local resolver. If the array contains the string ".", it

indicates the network requires filtering for all domains. If "requireDNSFiltering" is present with a boolean value of true, the network is indicating that it expects all client systems to send the names indicated by "dnsFilteredZones" to the local resolver. If "requireDNSFiltering" is not present or set to false, then the filtering service is considered to be optional for clients that want to use it as a service to enforce desired policy.

Clients that receive indication of filtering requirements SHOULD NOT use any other resolver for the filtered domains, but treat the network as claiming authority. However, since this filtering cannot be authenticated, this behavior SHOULD NOT be done silently without user consent.

Networks that try to interfere with connections to encrypted DNS resolvers without indicating a requirement for filtering cannot be distinguished from misconfigurations or network attacks. Clients MAY choose to avoid sending any user-initiated connections on such networks to prevent malicious interception.

7. Performance Considerations

One of the challenges of using non-local DNS servers (such as cloud-based DoH servers) is that recursive queries made by these servers will originate from an IP address that is not necessarily geographically related to the client. Many DNS servers make assumptions about the geographic locality of clients to their recursive resolvers to optimize answers. To avoid this problem, the client's subnet can be forwarded to the authoritative server by the recursive using the EDNS0 Client Subnet feature. Oblivious DoH discourages this practice for privacy reasons. However, sharing this subnet, while detrimental to privacy, can result in better targeted DNS resolutions.

Adaptive DNS splits DoH queries into two sets: those made to Designated DoH Servers, and those made to Oblivious DoH servers. Oblivious queries are sensitive for privacy, and can encounter performance degradation as a result of not using the client subnet. Queries to designated DoH servers, on the other hand, are sent directly by clients, so the client IP address is made available to these servers. Since these servers are designated by the authority for the names, they can use the IP address subnet information to tune DNS answers.

Based on these properties, clients SHOULD prefer lookups via Designated DoH Servers over oblivious mechanisms whenever possible. Servers can encourage this by setting large TTLs for SVCB records and using longer TTLs for responses returned by their Designated DoH

Server endpoints which can be more confident they have accurate addressing information.

8. Security Considerations

In order to avoid interception and modification of the information retrieved by clients using Adaptive DNS, all exchanges between clients and servers are performed over encrypted connections, e.g., TLS.

Malicious adversaries may block client connections to all DoH or Oblivious DoH services as a Denial-of-Service (DoS) measure. Clients which cannot connect to any proxy may, by local policy, fall back to unencrypted DNS if this occurs.

9. Privacy Considerations

Clients must be careful in determining to which DoH servers they send queries directly without proxying. A malicious DoH server that can direct queries to itself can track or profile client activity. In order to avoid the possibility of a spoofed SVCB record designating a malicious DoH server for a name, clients MUST ensure that such records validate using DNSSEC [RFC4033].

Even servers that are officially designated can risk leaking or logging information about client lookups. Such risk can be mitigated by further restricting the list of DoH servers that are whitelisted for direct use based on client policy.

Using Oblivious DoH reduces the risk that a single DoH server can track or profile a client. However, clients should exercise caution when using Oblivious DoH responses from resolvers that do not carry DNSSEC signatures. An adversarial Oblivious Target resolver that wishes to learn the IP address of clients requesting resolution for sensitive domains can redirect clients to unique addresses of its choosing. Clients that use these answers when establishing TLS connections may then leak their local IP address to chosen server. Thus, when Oblivious DoH answers are returned without DNSSEC, Privacy-Sensitive Connections concerned about this attack SHOULD conceal their IP address via a TLS- or HTTP-layer proxy or some other tunneling mechanism.

An adversary able to see traffic on each path segment of a DoH or Oblivious DoH query (e.g., from client to proxy, proxy to target, and target to an authoritative DNS server) can link queries to specific clients with high probability. Failure to observe traffic on any one of these path segments makes this linkability increasingly difficult. For example, if an adversary can only observe traffic between a

client and proxy and egress traffic from a target, then it may be difficult identify a specific client's query among the recursive queries generated by the target.

10. IANA Considerations

10.1. DoH Template PvD Key

This document adds a key to the "Additional Information PvD Keys" registry [I-D.ietf-intarea-provisioning-domains].

JSON key	Description	Type	Example
dohTemplate	DoH URI Template [RFC8484]	String	"https://dnsserver.example.net/dns-query{?dns}"

10.2. DNS Filtering PvD Keys

This document adds a key to the "Additional Information PvD Keys" registry [I-D.ietf-intarea-provisioning-domains].

JSON key	Description	Type	Example
requireDNSFiltering	A flag to indicate that the network requires filtering all DNS traffic using the provisioned resolver.	Boolean	true
dnsFilteredZones	A list of DNS domains as strings that represent domains that can be filtered by the provisioned resolver.	Array of Strings	["."]

Any network that sets the "requireDNSFiltering" boolean to false but provides "dnsFilteredZones" advertises the optional service of filtering on the provisioned network.

An "." in the "dnsFilteredZones" array represents a wildcard, which can be used to indicate that the network is requesting to filter all

names. Any more specific string represents a domain that requires filtering on the network.

10.3. DoH URI Template DNS Parameter

If present, this parameters indicates the URI template of a DoH server that is designated for use with the name being resolved. This is a string encoded as UTF-8 characters.

Name: dohuri

SvcParamKey: TBD

Meaning: URI template for a designated DoH server

Reference: This document.

11. Acknowledgments

Thanks to Erik Nygren, Lorenzo Colitti, Tommy Jensen, Mikael Abrahamsson, Ben Schwartz, Ask Hansen, Leif Hedstrom, Tim McCoy, Stuart Cheshire, Miguel Vega, Joey Deng, Ted Lemon, and Elliot Briggs for their feedback and input on this document.

12. References

12.1. Normative References

[I-D.ietf-intarea-provisioning-domains]
Pfister, P., Vyncke, E., Pauly, T., Schinazi, D., and W. Shao, "Discovering Provisioning Domain Names and Data", draft-ietf-intarea-provisioning-domains-08 (work in progress), October 2019.

[I-D.ietf-tls-esni]
Rescorla, E., Oku, K., Sullivan, N., and C. Wood, "Encrypted Server Name Indication for TLS 1.3", draft-ietf-tls-esni-04 (work in progress), July 2019.

[I-D.irtf-cfrg-hpke]
Barnes, R. and K. Bhargavan, "Hybrid Public Key Encryption", draft-irtf-cfrg-hpke-00 (work in progress), July 2019.

[I-D.nygren-dnsop-svcb-httpssvc]
Schwartz, B., Bishop, M., and E. Nygren, "Service binding and parameter specification via the DNS (DNS SVCB and HTTPSSVC)", draft-nygren-dnsop-svcb-httpssvc-00 (work in progress), September 2019.

[I-D.pauly-dprive-oblivious-doh]
Kinnear, E., Pauly, T., Wood, C., and P. McManus, "Oblivious DNS Over HTTPS", draft-pauly-dprive-oblivious-doh-00 (work in progress), October 2019.

[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>.

[RFC7231] Fielding, R., Ed. and J. Reschke, Ed., "Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content", RFC 7231, DOI 10.17487/RFC7231, June 2014, <https://www.rfc-editor.org/info/rfc7231>.

[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>.

[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>.

12.2. Informative References

[I-D.ietf-capport-architecture]
Larose, K. and D. Dolson, "CAPPORT Architecture", draft-ietf-capport-architecture-04 (work in progress), June 2019.

[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>.

[RFC2132] Alexander, S. and R. Droms, "DHCP Options and BOOTP Vendor Extensions", RFC 2132, DOI 10.17487/RFC2132, March 1997, <https://www.rfc-editor.org/info/rfc2132>.

[RFC8106] Jeong, J., Park, S., Beloeil, L., and S. Madanapalli, "IPv6 Router Advertisement Options for DNS Configuration", RFC 8106, DOI 10.17487/RFC8106, March 2017, <https://www.rfc-editor.org/info/rfc8106>.

[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>.

Authors' Addresses

Eric Kinnear
Apple Inc.
One Apple Park Way
Cupertino, California 95014
United States of America

Email: ekinnear@apple.com

Tommy Pauly
Apple Inc.
One Apple Park Way
Cupertino, California 95014
United States of America

Email: tpauly@apple.com

Chris Wood
Apple Inc.
One Apple Park Way
Cupertino, California 95014
United States of America

Email: cawood@apple.com

Patrick McManus
Fastly

Email: mcmanus@ducksong.com