Workgroup: Network Working Group
Internet-Draft:
draft-pauly-add-requirements-00
Published: 21 August 2020
Intended Status: Informational
Expires: 22 February 2021
Authors: T. Pauly
Apple Inc.
E. Kinnear
Apple Inc.
C.A. Wood
Cloudflare
P. McManus
Fastly
T. Jensen
Microsoft

Adaptive DNS Discovery Requirements

Abstract

This document describes several use cases for discovering DNS resolvers that support encrypted transports, and discusses how solutions for these use cases can be designed to use common mechanisms. It also considers the requirements for privacy and security when designing resolver discovery mechanisms.

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 22 February 2021.

Table of Contents

1. Introduction

Several protocols for protecting DNS traffic with encrypted transports have been defined, such as DNS-over-TLS (DoT) [RFC7858] and DNS-over-HTTPS (DoH) [RFC8484]. Encrypted DNS can provide many security and privacy benefits for network clients.

While it is possible for clients to hard-code encrypted DNS resolvers to use, dynamic discovery and provisioning of encrypted resolvers can expand the usefulness and applicability of encrypted DNS to many more use cases.

This document first describes several use cases for discovering DNS resolvers that support encrypted transports (Section 2).

Next, it discusses how solutions for these use cases can be grouped and categorized to point to the usefulness of common mechanisms (Section 3).

Last, it considers the requirements for privacy and security when designing resolver discovery mechanisms (Section 4).

This document is designed to aid in discussion of the Adaptive DNS Discovery (ADD) working group as defines mechanism requirements.

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. Use Cases

This section describes various use cases for which it is possible to discover an encrypted resolver. For each use case, the privacy and security benefits of adding encrypted resolution are briefly described.

2.1. Network-provisioned resolvers

DNS servers are often provisioned by a network as part of DHCP options [RFC2132] or IPv6 Router Advertisement (RA) options [RFC8106]. These options describe one or more DNS resolver IP addresses, to be used for traditional unencrypted DNS.

Using an encrypted resolver that is provisioned by the network can provide several benefits that are not possible if only unencrypted DNS is used:

* Prevent other devices on the network from observing client DNS messages

* Verify that answers come from the selected DNS resolver

* Authenticate that the DNS resolver is the one provisioned by the network

Often, network-provisioned resolvers are forwarders running on a local router. The discovered encrypted resolvers in these cases may either be local fowarders themselves, or an associated resolver that is in the network (thus bypassing the router's DNS forwarder).

2.2. Client-selected resolvers

Client devices often allow a user or administrator to select a specific DNS resolver to use on certain networks, or on all networks. Historically, this selection was specified only with an IP address.

Discovering if the selected resolver supports encryption, along with the configuration for the encrypted resolver, allows the client to "upgrade" connections to use encrypted DNS. This can provide several benefits:

* Prevent devices along the network path to the selected resolver from observing client DNS messages

* Verify that answers come from the selected DNS resolver

* Authenticate that the DNS resolver is the one selected by the client

2.3. VPN resolvers

Virtual Private Networks (VPNs) also can provision DNS resolvers. In addition to being able to use DHCP or RAs, VPNs can provision DNS information in an explicit configuration message. For example, IKEv2 can provision DNS servers using Configuration Attributes [RFC7296].

VPNs can also configure Split DNS rules to limit the use of the configured resolvers to specific domain names [RFC8598].

Discovering an encrypted resolver that is provisioned by a VPN can provide the same benefits as doing so for a local network, but applied to the private network. When using Split DNS, it becomes possible to use a one encrypted resolver for private domains, and another for other domains.

2.4. Encrypted resolvers for private names

Similar to how VPN DNS configurations can use Split DNS for private names, other network environments can support resolution of private names. For example, an enterprise-managed Wi-Fi network might be able to access both the Internet an a private intranet. In such a scenario, the private domains managed by the enterprise might only be resolvable using a specific DNS resolver.

Discovering an encrypted resolver for private domains allows a client to perform Split DNS while maintaining the benefits of encrypted DNS. For example, a client could use a client-selected encrypted resolver for most domains, but use a different encrypted resolver for enterprise-private domains.

This has the privacy benefit of only exposing DNS queries to the enterprise that fall within a limited set of domains, if there is a more preferred option for generic Internet traffic.

Using encrypted DNS for private names also opens up the possibility of doing private name resolution outside of the content of a VPN or managed network. If the DNS resolver authenticates clients, it can offer its resolver for private names on a publicly accessible server, while still limiting the visibility of the DNS traffic.

2.5. Encrypted resolvers for local or home content

Accessing locally-hosted content can require the use of a specific resolver. For example, captive networks or networks with walled-garden content like media on airplane Wi-Fi networks can rely on using a resolver hosted on the local network.

In cases where a client is using an encrypted resolver provisioned by a network, and that encrypted resolver is able to resolve names local content, this can fall into the use case described in Section

2.1. However, it might be necessary to discover a local encrypted resolver along with specific domains if:

* the network-provisioned encrypted resolver is not able to resolve local-only names, or

* the client has a more-preferred encrypted resolver for generic traffic, and would otherwise not be able to access local content

This case also include accessing content specific to a home network.

2.6. Encrypted resolvers for content providers

Content Delivery Networks (CDNs), and content-providers more broadly, can also provide encrypted DNS resolvers that can be used by clients over the public Internet. These resolvers can either allow resolution of all public names (like normal recursive resolvers), or be designed to serve a subset of names managed by the content provider (like an authoritative resolver). Using these resolvers can allow the content provider to directly control how DNS answers are used for load balancing and address selection, which could improve performance of connections to the content provider.

Using a content-provider's encrypted resolver can also provide several privacy and security benefits:

* Prevent devices along the network path to the content-provider's resolver from observing client DNS messages

* Verify that answers come from the entity that manages the domains being resolved

* Reduce the number of entities able to monitor the specific names accessed by a client to only the client and the content provider, assuming that the content provider would already see the names upon a secure connection later being made based on the DNS answers (e.g., in the TLS SNI extension)

3. Discovery mechanisms

The use cases described in Section 2 do not all necessarily require separate mechanisms.

Generally, the use cases can be summarized in two categories:

1. Resolver upgrade: Discover encrypted resolvers equivalent to (or associated with) unencrypted resolvers. Examples include network-provisioned, client-selected, and VPN-configured resolvers.

2. Domain-specific resolvers: Discover encrypted resolvers applicable to a limited set of domains. Examples include resolvers for enterprise or private names, local content, and CDN content.

Resolver upgrade mechanisms can either add new parameters to existing provisioning mechanisms (adding necessary information to use DoT or DoH to options in DHCP, RAs, or IKEv2) or else provide a way to communicate with a provisioned unencrypted DNS resolver and discover the equivalent or associated encrypted DNS resolver.

Domain-specific resolver discovery mechanisms additionally need to provide some information about the applicability and capabilities of encrypted resolvers. This information can either be provisioned or can be discovered based on clients actively trying to access content.

4. Privacy and security requirements

Encrypted DNS improves the privacy and security of DNS queries and answers in the presence of malicious attackers. Such attackers are assumed to interfere with or otherwise impede DNS traffic and corresponding discovery mechanisms. They may be on-path or off-path between the client and entities with which the client communicates [RFC3552]. These attackers can inject, tamper, or otherwise interfere with traffic as needed. Given these capabilities, an attacker may have a variety of goals, including, though not limited to:

- Monitor and profile clients by observing unencrypted DNS traffic
- Modify unencrypted DNS traffic to filter or augment the user experience
- Block encrypted DNS

Clients cannot assume that their network does not have such an attacker unless given some means of authenticating or otherwise trusting the communication with their DNS resolver.

Given this type of attacker, resolver discovery mechanisms must be designed carefully to not worsen a client's security or privacy posture. In particular, attackers must not be able to:

- Redirect DNS traffic to themselves.
- Override or interfere with the resolver preferences of a user or administrator.
- Cause clients to use a discovered resolver which has no authenticated delegation from a client-known entity.
- Influence automatic discovery mechanisms such that a client uses one or more resolvers that are not otherwise involved with providing service to the client, such as: a network provider, a VPN server, a content provider being accessed, or a server that the client has manually configured.

Beyond these requirements, standards describing resolver discovery mechanisms must not place any requirements on clients to select particular resolvers over others.

4.1. On opportunistic encryption

Opportunistic encrypted DNS, when the client cannot authenticate the entity that provides encrypted DNS, does not meet the requirements laid out here for resolver discovery. While opportunistic encryption can provide some benefits, specifically in reducing the ability for other entities to observe traffic, it is not a viable solution against an on-path attacker.

Performing opportunistic encrypted DNS does not require specific discovery mechanisms. Section 4.1 of [RFC7858] already describes how to use DNS-over-TLS opportunistically.

4.2. Handling exceptions and failures

Even with encrypted DNS resolver discovery in place, clients must be prepared to handle certain scenarios where encrypted DNS cannot be used. In these scenarios, clients must consider if it is appropriate to fail open by sending the DNS queries without encryption, fail closed by not doing so, or presenting a choice to a user or administrator. The exact behavior is a local client policy decision.

Some networks that use Captive Portals will not allow any Internet connectivity until a client has interacted with the portal [I-D.ietf-capport-architecture]. If these networks do not use encrypted DNS for their own resolution, a client will need to perform unencrypted DNS queries in order to get out of captivity. Many operating systems have specific client code responsible for detecting and interacting with Captive Portals; these system components may be good candidates for failing open, since they do not generally represent user traffic.

Other networks may not allow any use of encrypted DNS, or any use of encrypted DNS to resolvers other than a network-provisioned resolver. Clients should not silently fail open in these cases, but if these networks are trusted by or administered by the user, the user may want to specifically follow the network's DNS policy instead of what the client would do on an unknown or untrusted network.

5. Informative References

[I-D.ietf-capport-architecture]
Larose, K., Dolson, D., and H. Liu, "Captive Portal Architecture", Work in Progress, Internet-Draft, draft-ietf-capport-architecture-09, 8 August 2020, <http://www.ietf.org/internet-drafts/draft-ietf-capport-architecture-09.txt>.

[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>.

[RFC3552] Rescorla, E. and B. Korver, "Guidelines for Writing RFC Text on Security Considerations", BCP 72, RFC 3552, DOI 10.17487/RFC3552, July 2003, <https://www.rfc-editor.org/info/rfc3552>.

[RFC7296] Kaufman, C., Hoffman, P., Nir, Y., Eronen, P., and T. Kivinen, "Internet Key Exchange Protocol Version 2 (IKEv2)", STD 79, RFC 7296, DOI 10.17487/RFC7296, October 2014, <https://www.rfc-editor.org/info/rfc7296>.

[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>.

[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>.

[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>.

[RFC8598] Pauly, T. and P. Wouters, "Split DNS Configuration for the Internet Key Exchange Protocol Version 2 (IKEv2)", RFC 8598, DOI 10.17487/RFC8598, May 2019, <https://www.rfc-editor.org/info/rfc8598>.

Authors' Addresses

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

Email: tpauly@apple.com

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

Email: ekinnear@apple.com

Christopher A. Wood
Cloudflare
101 Townsend St
San Francisco,
United States of America

Email: caw@heapingbits.net

Patrick McManus
Fastly

Email: mcmanus@ducksong.com

Tommy Jensen
Microsoft

Email: tojens@microsoft.com