Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: April 26, 2019

P. Hoffman
ICANN

October 23, 2018

Associating a DoH Server with a Resolver
draft-hoffman-resolver-associated-doh-04

Abstract

Browsers and web applications may want to know if there are one or more DoH servers associated with the DNS recursive resolver that the operating system is already using. This would allow them to get DNS responses from a resolver that the user (or, more likely, the user's network administrator) has already chosen. This document describes a protocol for a resolver to tell a client what its associated DoH servers are.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

DoH [RFC8484] requires that one or more DoH servers be configured for the DoH client. That document does not say how the DoH servers are found, nor how to select from a list of possible DoH servers, nor what the user interface (UI) for the configuration should be.

There is a use case for browsers and web applications who have one or more currently-configured DNS recursive resolvers wanting to use DoH for DNS resolution instead. (In the rest of this document "browsers and web applications" are just called "applications".) For example, the recursive resolver knows how to give correct answers to DNS queries that contain names that are only resolvable in the local context. Users typically configure their DNS recursive resolvers with through manual configuration (such as manually editing a /etc/named.conf file) or through automatic configuration from a protocol such as DHCP.

The client that wants to change from its currently-configured Do53 recursive resolver(s) to one or more DoH servers might be the stub resolver in an operating system, although at this time it is rare that such stub resolvers can use DoH. A much more likely use case is an application that is getting name resolution through the stub resolver on the computer on which it is running. The user of the application might have a preference for using a DoH server, and they might need to use a DoH server that is associated with the resolver

that the computer is currently using so that its queries for non-global names are answered correctly. They may also be required to use only resolvers that are approved by their organization's network operators.

To address these use cases, this document defines a new special use domain name (described in [RFC6761]) and a well-known URI [I-D.nottingham-rfc5785bis]. When combined, they allow an application that can use the POSIX "getaddrinfo()" function and resolve HTTP and HTTPS URLs to get a list of the DoH servers associated with at least one of the resolvers being used by the operating system on the system on which the application is being run.

It is important to note that using a DoH server based on the protocol defined in this document will currently result in communicating with opportunistic encryption [RFC7435] using "unauthenticated, encrypted communication" instead of "authenticated, encrypted communication". This is covered in more detail in Section 8.

The design choices for this protocol, particularly earlier designs that were deemed unusable, are described in Section 5.

2. Terminology

In this document, the combination of "browsers and web applications" is just called "applications".

In this document, "DoT" is used to indicate DNS over TLS as defined in [RFC7858].

In this document, "Do53" is used to indicate DNS over UDP or TCP as defined in [RFC1035].

"DoH client" and "DoH server" are defined in [RFC8484].

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. Finding the DoH Servers Associated with a Resolver

To find the DoH Servers associated with a resolver, an application uses a special use domain name that causes a resolver to return its IP addresses. It uses those IP addresses as part of a well-known URI to find out the URI templates [RFC6570] to use for the DoH server(s) associated with the resolver.

3.1. Step 1: Finding the IP Addresses of a Resolver

An application is able to use the POSIX "getaddrinfo()" function to convert host names into IP addresses through the stub resolver in the operating system on which it is running. It can also send queries to a resolver, but it would need to have the address of that resolver first.

In order for an application to find the address of the resolver that the operating system is using, it uses the POSIX "getaddrinfo()" function (or some equivalent) with the special use name "resolver-addresses.arpa". When a resolver that understands this special use domain name receives a query for either resolver-addresses.arpa/IN/A or resolver-addresses.arpa/IN/AAAA, it returns its own IP addresses in the answer.

3.2. Step 2: Finding the DoH Servers Associated with a Resolver

To find the DoH servers associated with a resolver, the client uses the addresses returned from the query to resolver-addresses.arpa and sends a query to:

https://ADDRESS/.well-known/doh-servers-associated/

where "ADDRESS" is an IP address discovered in Section 3.1.

The resolver replies with a list of its associated DoH servers as URI Templates [RFC6570].

[[Need to describe the media type; likely JSON; and the list specifics.]]

Note that the well-known URL above uses the HTTPS scheme and no port number. A resolver using the protocol defined in this document MUST provide HTTP over TLS on port 443 as defined in [RFC2818].

A resolver that implements this protocol but has no DoH servers associated with it returns an empty list.

The result of Section 3.1 may be a list of more than one IP addresses. This document does not define a way for an application to choose between multiple IP addresses. For example, the application might try all the IP addresses, or try them in random order until it gets a result, and so on.

The result of resolving the well-known URI can be a list of more than one URI templates, possibly pointing resources in very different places on the Internet. This document does not define a way for the

application to choose which DoH servers to use if presented with multiple choices.

An application that is willing to use opportunistic encryption as defined in [RFC7435] MAY ignore authentication failures when resolving the well-known URL.

An application that is not willing to use opportunistic encryption as defined in [RFC7435] MUST NOT ignore authentication failures when going to the well-known URL. However, as described in Section 8, such an application is unlikely to be able to exist today.

[[Need to talk about HTTP caching]]

A client MUST try to establish a new list of DoH servers associated with a resolver every time the configured resolver in the operating system changes.

4. User Interface

For this protocol to be useful in an application, the application needs to have an entry in its configuration interface where the allowed DoH servers are listed that indicates that a DoH server from the configured Do53 or DoT resolver is allowed. That wording might say something like "DoH server associated with my current resolver".

This is a place where browsers and web applications are different. Most browsers have configuration interfaces, while most web applications do not.

5. Design Choices

The primary use case for this protocol is an application that is getting name resolution through the stub resolver on the computer on which it is running wanting to switch its name resolution to DoH. A secondary use case is an OS that wants to make a similar switch.

An earlier design suggestion was to use a new RRtype with a query to ./IN/NEWRRTYPE. However, it was pointed out that this would not work going through stub resolvers that validate DNSSEC.

An earlier design suggestion was to use DHCP to tell the OS the DoH servers that the stub resolver might use. That protocol is orthogonal to the one in this document in that it addresses a different use case. If both the protocol in this document and a DHCP-based protocol are standardized, they could co-exist. However, there is no current mechanism for a stub resolver to tell an application what DoH server the stub resolver is using, so DoH

configuration in the stub resolver would not prevent the application from trying to find a DoH server on its own.

An earlier design suggestion was to use an EDNS0 [RFC6891] extension. The design chosen in this document meets the use case better because applications cannot communicate EDNS0 extensions to the stub resolver.

An earlier design suggestion used a special use domain name of resolver-associated-doh.arpa with an RRtype of TXT. The design chosen in this document meets the use case better because applications cannot query the stub resolver for types other than address records.

6. IANA Considerations

IANA will record the domain name "resolver-addresses.arpa." in the "Special-Use Domain Names" registry [SUDN]. IANA MUST NOT delegate resolver-addresses.arpa in the .arpa zone.

[[When this document settles down, need to register ".well-known/doh-servers-associated" as specified in [I-D.nottingham-rfc5785bis].]]

7. Privacy Considerations

Allowing an application to use DoH instead of Do53 increases communication privacy because of the TLS protection, even if that communication is unauthenticated. If the communication is unauthenticated (which it will be using current technologies; see Section 8), the communication between the application and the DoH server to be private from anyone other than a on-path attacker.

When a Do53 or DoT server indicates that a particular DoH server is associated with it, the application might assume that the DoH server has the same information privacy policies as the Do53 or DoT server. Therefore, a Do53 or DoT server SHOULD NOT recommend a DoH server unless that DoH server has the same (or better) information privacy policy as the Do53 or DoT server.

8. Security Considerations

[RFC7435] defines "unauthenticated, encrypted communication" and "authenticated, encrypted communication". Those definitions make it clear that authentication is needed in every step in order to consider communication authenticated and encrypted.

There is currently no way for an application to know whether the operating system's stub resolver is using a transport that assures data integrity such as DoT. This means that the protocol in Section 3.1 is not authenticated. In the future, such a signal might be defined and deployed, but until then, the lack of assurance of authentication in the first step of this protocol (getting the resolver's IP address) means that the result will always be unauthenticated.

Even is an application could determine the use of a transport like DoT for Section 3.1, the application would also need to know whether the transport was authenticated or was simply chosen opportunistically. Thus, if in the future, a signal about the DNS transport being used by the stub resolver might be defined and deployed, that signal would also have to specify if the transport is also authenticated.

The protocol defined in Section 3.2 explicitly allows ignoring the authentication of the results of resolving the well-known URI. Doing so of course causes the result to be unauthenticated, encrypted communication.

9. References

9.1. Normative References

[I-D.nottingham-rfc5785bis]
Nottingham, M., "Well-Known Uniform Resource Identifiers (URIs)", draft-nottingham-rfc5785bis-08 (work in progress), October 2018.

[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>.

[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>.

[RFC2818] Rescorla, E., "HTTP Over TLS", RFC 2818, DOI 10.17487/RFC2818, May 2000, <https://www.rfc-editor.org/info/rfc2818>.

[RFC6570] Gregorio, J., Fielding, R., Hadley, M., Nottingham, M., and D. Orchard, "URI Template", RFC 6570, DOI 10.17487/RFC6570, March 2012, <https://www.rfc-editor.org/info/rfc6570>.

[RFC7435] Dukhovni, V., "Opportunistic Security: Some Protection Most of the Time", RFC 7435, DOI 10.17487/RFC7435, December 2014, <https://www.rfc-editor.org/info/rfc7435>.

[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>.

[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>.

[SUDN] "Special-Use Domain Names", n.d., <https://www.iana.org/assignments/special-use-domain-names/>.

9.2. Informative References

[RFC6761] Cheshire, S. and M. Krochmal, "Special-Use Domain Names", RFC 6761, DOI 10.17487/RFC6761, February 2013, <https://www.rfc-editor.org/info/rfc6761>.

[RFC6891] Damas, J., Graff, M., and P. Vixie, "Extension Mechanisms for DNS (EDNS(0))", STD 75, RFC 6891, DOI 10.17487/RFC6891, April 2013, <https://www.rfc-editor.org/info/rfc6891>.

Acknowledgments

The use case in this document was inspired by discussions and the DRIU BoF at IETF 102 and later in the DNSOP Working Group. Vladimir Cunat, Philip Homburg, and Shumon Huque offered useful advice to greatly improve an early version of the protocol.

Author's Address

Paul Hoffman
ICANN

Email: paul.hoffman@icann.org