Workgroup: ADD
Internet-Draft:
draft-jt-add-dns-server-redirection-01
Published: 12 March 2023
Intended Status: Standards Track
Expires: 13 September 2023
Authors: J. Todd T. Jensen C. Mosher
Quad9 Microsoft Quad9

Handling Encrypted DNS Server Redirection

Abstract

This document defines Encrypted DNS Server Redirection (EDSR), a mechanism for encrypted DNS servers to redirect clients to other encrypted DNS servers. This enables dynamic routing to geo-located or otherwise more desirable encrypted DNS servers without modifying DNS client endpoint configurations or the use of anycast by the DNS server.

About This Document

This note is to be removed before publishing as an RFC.

The latest revision of this draft can be found at https://example.com/LATEST. Status information for this document may be found at https://datatracker.ietf.org/doc/draft-jt-add-dns-server-redirection/.

Discussion of this document takes place on the WG Working Group mailing list (mailto:add@ietf.org), which is archived at https://example.com/WG. Subscribe at https://www.ietf.org/mailman/listinfo/add/.

Source for this draft and an issue tracker can be found at https://github.com/johnhtodd/draft-DOH-redirect.

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 13 September 2023.

Copyright Notice

Table of Contents

1. Conventions and Definitions
2. Introduction
3. DNS client behavior
   3.1. Discovering redirections
   3.2. Refreshing redirections
   3.3. Multiple redirections
   3.4. Network changes
4. DNS server behavior
5. Deployment Considerations
   5.1. Large trees of redirections
   5.2. Redirection TTLs
   5.3. Including IP addresses in EDSR responses
6. Security Considerations
   6.1. Trusting the redirected connection
   6.2. Use with unencrypted DNS
7. Privacy Considerations
8. Data Flow Considerations
   8.1. Data Scope
   8.2. Data Visibility
   8.3. Data centralization
9. IANA Considerations
10. References
   10.1. Normative References
   10.2. Informative References
Appendix A. Acknowledgments
Authors' Addresses

1. Conventions and Definitions

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. Introduction

Encrypted DNS Server Redirection (EDSR) is a protocol that allows an encrypted DNS resolver whose configuration is well known to clients to redirect them to other, more desirable resolvers without having to support anycast and without having to configure clients with these other resolvers ahead of time. It uses the mechanism defined by DDR [I-D.ietf-add-ddr] to redirect an encrypted DNS client from one encrypted DNS resolver to another encrypted DNS resolver. Where DDR uses a threat model that presumes the initial DNS traffic is unencrypted, EDSR applies when the initial DNS traffic is already encrypted.

One example of what makes redirection to another resolver desirable is geolocation. A DNS service may document one or a few well known resolver configurations even though it routes traffic to hundreds or thousands of resolvers that are closer to the client, reducing latency and making DNS resolutions more applicable to the client.

3. DNS client behavior

3.1. Discovering redirections

When a DNS client first opens a connection to an encrypted DNS server, it **MUST** send a SVCB query for the name of the resolver to discover its encrypted DNS configuration. The DNS client **SHOULD** open a connection to the server returned in the SVCB query using the TargetName and one of the IP addresses returned in additional A/AAAA records for the same name. Once a connection has been successfully opened, as subsequently described by reaching a suitable server at the end of the redirection chain, the client **SHOULD** close the first connection.

If the returned SVCB record indicates a server with the same domain name as the current encrypted DNS connection, even if it contains different values in additional A or AAAA records, or different values in the ipv4hint or ipv6hint fields, then the redirection is considered to be from the server to itself. Clients **SHOULD NOT** follow these redirections generally. However, clients receiving preferable encryption parameters as part of the SVCB response **MAY** choose to reconnect to negotiate to upgrade to the preferred encryption method. When doing so, there is no need for the client to

repeat EDSR as the redirection from the server to itself has terminated the redirection chain.

The client does not need to wait for the results of the redirection discovery query before sending other DNS queries on the connection, though they **SHOULD** gracefully close the connection as soon as it has successfully established a connection to the server it was redirected to and received or timed out the outstanding queries on the original connection.

See the considerations section for reasons a client **MAY** choose to decline a redirection.

3.2. Refreshing redirections

EDSR allows a client to be redirected from an encrypted DNS resolver it was somehow configured to use. When the redirection TTL expires, the client **SHOULD** return to using its originally configured server unless it can refresh the redirection beforehand. This allows the client to honor the intention of whatever configuration method was used to instruct it to use the original encrypted DNS resolver.

If a chain of redirections was followed, the effective TTL of the redirection is the minimum of the TTLs encountered along the chain. Clients **SHOULD** however cap this value to some minimum value at their discretion to avoid frequent redirection checking when latency plus an incidentally low TTL along the chain results in near-zero effective TTLs.

3.3. Multiple redirections

When clients receive more than one valid SVCB response, they **SHOULD** prefer using the redirections that match their configuration (such as supported IP address family or desired encrypted DNS protocol) in ascending order of the SVCB priority. Once a successful connection is made to a redirected destination, clients **MAY** choose to discard other results in favor of restarting EDSR with the originally configured resolver.

Redirections are considered to be a one-to-one relationship (starting with one recursive resolver and following its redirections should result in one replacement recursive resolver). It is not expected that a stub resolver ends up using more recursive resolvers than it was originally configured with when using EDSR.

3.4. Network changes

When a client device changes what network it is connected to, it **SHOULD** forget pre-existing redirections and start EDSR over with the originally configured resolvers. This ensures that a resolver which

redirects clients based on their source network can behave accordingly.

Note that this is unrelated to what resolvers a client is originally configured with. For example, a client which is configured to always used the resolvers advertised by DHCP will likely start with different original resolvers when changing networks. How a client is configured with DNS resolvers is out of scope for this document. EDSR only provides a mechanism for clients to discover redirections from resolvers they were previously configured to use.

4. DNS server behavior

DNS resolvers who want to redirect clients to other resolvers **MUST** respond to SVCB [I-D.ietf-add-svcb-dns] queries for their own domain names with records that describe the configuration of the destination server. Servers **SHOULD** be prepared for clients to not follow the redirection immediately as connection failures or other issues may lead to clients being unable to follow the redirection. Servers who are redirecting due to being overloaded **MAY** respond as they normally would to overwhelming traffic.

Guidance in Section 5 of [I-D.ietf-dnsop-svcb-https] to improve performance by including additional A/AAAA records with the SVCB response **SHOULD** be followed.

Redirections **MUST** only redirect to resolvers which support at least the same protocol, address family, port, and TLS minimum versions as the referring resolver. This ensures that redirections do not lead clients to resolvers that are not compatible with the client. In addition, servers **SHOULD** avoid redirecting to servers which will also redirect clients unless they are actively coordinating to ensure a positive client experience. See the Deployment Considerations section for more details.

5. Deployment Considerations

5.1. Large trees of redirections

It is possible for DNS servers to redirect clients to DNS servers which also redirect clients. Clients which are presented with long chains of redirections **MAY** choose to stop following redirections to reduce connection thrashing. DNS server operators **SHOULD** deploy redirection behavior mindfully to avoid long chains of redirection.

Servers **SHOULD** ensure their redirections do not create loops, where clients are redirected to a server it already encountered earlier in the process. Clients **MAY** stop following redirections when they detect this, but may also take a simpler approach, following only a maximum number of redirections.

5.2. Redirection TTLs

Servers **SHOULD** provide sufficiently long TTLs for clients to avoid the need to constantly repeat EDSR queries. Server operators should be mindful of redirection chains because unless they collaboratively control the TTLs of one another's redirections, redirection chains will end up with greatly reduced effective TTLs because the client will always use the lowest.

5.3. Including IP addresses in EDSR responses

If a recursive resolver does not include additional A/AAAA records per Section 5 of [I-D.ietf-dnsop-svcb-https], stub resolvers might end up failing the redirection if the redirection destination name cannot be resolved. Additionally, the recursive resolver **SHOULD** ensure records conntaining the same IP version as the existing connection are returned (if the stub is currently connected over IPv4, one or more A records **SHOULD** be included, and if the stub is currently connected over IPv6, one or more AAAA records **SHOULD** be included).

6. Security Considerations

6.1. Trusting the redirected connection

EDSR does not provide novel authentication or security mechanisms. Redirection is trusted by virtue of the server authentication via PKI through TLS [RFC5280]. The DNS stub resolver implementing EDSR **SHOULD** use whatever policies it uses for other TLS connections for encrypted DNS traffic to determine if a given TLS cert chain is trustworthy before proceeding with EDSR.

EDSR **MUST NOT** be used with encrypted DNS protocols that are not based on TLS. This scenario will require future standards work.

EDSR should not introduce any additional security considerations beyond use of the original encrypted resolver prior to redirection. Because the original connection was trusted, information sent over it about a new connection to use should be as trusted. This is analogous to the use of 3xx codes in HTTP to redirect HTTP clients to other servers. However, clients that wish to time bound vulnerabilities to attackers who compromise the original resolver **MAY** choose to implement a maximum TTL to honor on SVCB records that redirect to other servers.

6.2. Use with unencrypted DNS

EDSR **MUST NOT** be used to redirect unencrypted DNS traffic to any other resolver. This use case is called designation and is covered by Discovery of Designated Resolvers (DDR) as defined in

[I-D.ietf-add-ddr]. Not following DDR opens up a DNS client to malicious redirection to an attacker-controlled DNS server. For more information, see Section 7 of [I-D.ietf-add-ddr].

EDSR also **MUST NOT** be used to redirect encrypted DNS traffic to a resolver that advertises support for unencrypted DNS. This would reduce the security posture of the client. Clients **MUST NOT** follow an encrypted DNS redirection and then send unencrypted DNS traffic to the new resolver.

7. Privacy Considerations

A client **MAY** choose to not send other name queries until redirection is complete, but there should be no issue with sending queries to intermediate resolvers before redirection takes place. This is because the intermediate resolvers are considered to be appropriate destinations by the client even if the resolver wants to substitute another resolver for reasons other than name resolution results such as latency optimization or load balancing.

8. Data Flow Considerations

8.1. Data Scope

EDSR does not result in any additional data being shared by the end user, as the DNS queries going to the new resolver were already going to go to the original resolver.

8.2. Data Visibility

EDSR results in a 1:1 replacement of DNS resolvers used (future queries sent to the new resolver will not be sent to the original resolver anymore). This means the number of servers which see any given query remain the same.

This is only true if clients only use one redirected DNS server per original DNS server. If the DNS server offers more than one redirection, and the client validates and uses two or more of those redirections, then there will be greater data visibility (more destinations). This is however entirely within the client's choice following their own policy as a redundancy versus volume of exhausted data trade-off.

EDSR requires the redirection to another server to also use encrypted DNS, so no third-party will be introduced to the data flow unless the encryption is broken.

8.3. Data centralization

EDSR can only increase data centralization if multiple resolver operators choose to redirect DNS clients to the same, other DNS resolver. To prevent the reduction of their resolution redundancy, DNS clients **MAY** choose to ignore redirections if they find that they point to resolvers they are already configured to use, by a previous redirection or some other configuration.

9. IANA Considerations

This document has no IANA actions.

10. References

10.1. Normative References

[I-D.ietf-add-ddr] Pauly, T., Kinnear, E., Wood, C. A., McManus, P., and T. Jensen, "Discovery of Designated Resolvers", Work in Progress, Internet-Draft, draft-ietf-add-ddr-10, 5 August 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-add-ddr-10>.

[I-D.ietf-add-svcb-dns] Schwartz, B. M., "Service Binding Mapping for DNS Servers", Work in Progress, Internet-Draft, draft-ietf-add-svcb-dns-07, 11 August 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-add-svcb-dns-07>.

[I-D.ietf-dnsop-svcb-https] Schwartz, B. M., Bishop, M., and E. Nygren, "Service binding and parameter specification via the DNS (DNS SVCB and HTTPS RRs)", Work in Progress, Internet-Draft, draft-ietf-dnsop-svcb-https-12, 11 March 2023, <https://datatracker.ietf.org/doc/html/draft-ietf-dnsop-svcb-https-12>.

[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/rfc/rfc2119>.

[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/rfc/rfc8174>.

10.2. Informative References

[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/rfc/rfc5280>.

Appendix A. Acknowledgments

The authors would like to thank the following individuals for their invaluable feedback to this document: Ben Schwartz, Eric Orth, Erik Nygren, Ralph Weber, Ted Hardie, Tommy Pauly, Viktor Dukhovni, and Vittorio Bertola.

Authors' Addresses

J. Todd
Quad9

Email: jtodd@quad9.net

T. Jensen
Microsoft

Email: tojens@microsoft.com

C. Mosher
Quad9

Email: cmosher@quad9.net