Network Working Group
Internet-Draft
Intended status: Experimental
Expires: 14 October 2021

P. Hoffman
ICANN
P. van Dijk
PowerDNS

12 April 2021

Recursive to Authoritative DNS with Unauthenticated Encryption
draft-ietf-dprive-unauth-to-authoritative-00

Abstract

This document describes a use case and a method for a DNS recursive resolver to use unauthenticated encryption when communicating with authoritative servers. The motivating use case for this method is that more encryption on the Internet is better, and some resolver operators believe that unauthenticated encryption is better than no encryption at all. The method described here is optional for both the recursive resolver and the authoritative server. This method supports unauthenticated encryption using the same mechanism for discovery of encryption support for the server as [I-D.rescorla-dprive-adox-latest].

NOTE: The file name for this draft, draft-ietf-dprive-opportunistic-adotq, is now incorrect. This draft only covers unauthenticated encryption, not opportunistic encryption.

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 14 October 2021.

Copyright Notice

Table of Contents

1. Introduction 2
1.1. Use Case for Unauthenticated Encryption 3
1.2. Summary of Protocol 3
1.3. Definitions 4
2. Discovering Whether an Authoritative Server Uses Encryption 4
3. Resolving with Encryption 5
3.1. Resolver Session Failures 6
3.2. Resolver Process as Pseudocode 7
4. Serving with Encryption 7
5. Resolvers Reporting Errors to Authoritative Servers 8
6. IANA Considerations 8
7. Security Considerations 8
8. Acknowledgements 9
9. References 9
9.1. Normative References 9
9.2. Informative References 10
Authors' Addresses 11

1. Introduction

A recursive resolver using traditional DNS over port 53 may wish instead to use encrypted communication with authoritative servers in order to limit snooping of its DNS traffic by passive or on-path attackers. The recursive resolver can use unauthenticated encryption (defined in [RFC7435]) to achieve this goal.

This document describes the use case for unauthenticated encryption in recursive resolvers in Section 1.1. The encryption method with authoritative servers can be DNS-over-TLS [RFC7858] (DoT), DNS-over-HTTPS [RFC8484] (DoH), and/or DNS-over-QUIC [I-D.ietf-dprive-dnsoquic] (DoQ), as described in Section 3.

The document also describes a discovery method that shows if an authoritative server supports encryption in Section 2.

See [I-D.rescorla-dprive-adox-latest] for a description of the use case and a proposed mechanism for fully-authenticated encryption.

NOTE: The draft uses the SVCB record as a discovery mechanism for encryption by a particular authoritative server. Any record type that can show multiple types of encryption (currently DoT, DoH, and DoQ) can be used for discovery. Thus, this record type might change in the future, depending on the discussion in the DPRIVE WG.

1.1. Use Case for Unauthenticated Encryption

The use case in this document for unauthenticated encryption is recursive resolver operators who are happy to use encryption with authoritative servers if doing so doesn't significantly slow down getting answers, and authoritative server operators that are happy to use encryption with recursive resolvers if it doesn't cost much. In this use case, resolvers do not want to return an error for requests that were sent over an encrypted channel if they would have been able to give a correct answer using unencrypted transport.

Resolvers and authoritative servers understand that using encryption costs something, but are willing to absorb the costs for the benefit of more Internet traffic being encrypted. The extra costs (compared to using traditional DNS on port 53) include:

* Extra round trips to establish TCP for every session (but not necessarily for every query)
* Extra round trips for TLS establishment
* Greater CPU use for TLS establishment
* Greater CPU use for encryption after TLS establishment
* Greater memory use for holding TLS state

This use case is not expected to apply to all resolvers or authoritative servers. For example, according to [RSO_STATEMENT], some root server operators do not want to be the early adopters for DNS with encryption. The protocol in this document explicitly allows authoritative servers to signal when they are ready to begin offering DNS with encryption.

1.2. Summary of Protocol

This summary gives an overview of how the parts of the protocol work together.

* The resolver discovers whether any authoritative server of interest supports DNS with encryption by querying for the SVCB records [I-D.ietf-dnsop-svcb-https]. As described in [I-D.schwartz-svcb-dns], SVCB records can indicate that a server supports encrypted transport of DNS queries.

  NOTE: In this document, the term "SVCB record" is used _only_ for SVCB records that indicate encryption as described in [I-D.schwartz-svcb-dns]. SVCB records that do not have these indicators in the RDATA are not included in the term "SVCB record" in this document.

* The resolver uses any authoritative server with a SVCB record that indicates encryption to perform unauthenticated encryption.

* The resolver does not fail to set up encryption if the authentication in the TLS session fails.

1.3. Definitions

The terms "recursive resolver", "authoritative server", and "classic DNS" are defined in [I-D.ietf-dnsop-rfc8499bis].

"DNS with encryption" means transport of DNS over any of DoT, DoH, or DoQ. A server that supports DNS with encryption supports transport over one or more of DoT, DoH, or DoQ.

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. Discovering Whether an Authoritative Server Uses Encryption

A recursive resolver discovers whether an authoritative server supports DNS with encryption by looking for a cached SVCB record for the name of the authoritative server (with "_dns" prefix) with a positive answer. A cached SVCB record with a negative answer indicates that the authoritative server does not support any encrypted transport. Positive and negative responses for SVCB queries are cached the same way as for all other DNS resource records.

See [I-D.rescorla-dprive-adox-latest] for examples of querying for NS records and for SVCB records, and the interpretation of positive answers.

If the cache has no positive or negative answers for any SVCB record for any of a zone's authoritative servers, the resolver MAY send queries for the SVCB records for some or all of the zone's authoritative servers and wait for a positive response so that the resolver can use DNS with encryption for the original query. In this situation, the resolver MAY instead just use classic DNS for the original query but simultaneously queue queries for the SVCB records for some or all of the zone's authoritative servers so that future queries might be able to use DNS with encryption.

Discovery using SVCB records differs between resolvers using unauthenticated encryption and those using fully-authenticated encryption (described in [I-D.rescorla-dprive-adox-latest]). If the resolver is using unauthenticated encryption, the SVCB records do not need to be DNSSEC-signed.

DNSSEC validation of SVCB RRsets used strictly for this discovery mechanism is not mandated.

As described in [I-D.rescorla-dprive-adox-latest], these records may be in the resolver's cache because they came in the Additional section of a query for the NS records of a zone. This document does

not rely on that feature being standardized or operationally present to work.

Because some authoritative servers or middleboxes are misconfigured, requests for unknown RRtypes might be ignored by them. Resolvers should be ready to deal with timeouts or other bad responses to their SVCB queries.

3. Resolving with Encryption

A resolver following this protocol MUST use SVCB records in its cache to decide whether to use classic DNS or encryption to contact authoritative servers for a zone. If any of the SVCB records in the cache for the authoritative servers for a zone are positive responses, the resolver uses any of those servers for encryption. A resolver MUST NOT attempt encryption for a server that has a negative response in its cache for the associated SVCB record.

If all of the SVCB records for the authoritative servers in the cache for a zone are negative responses, the resolver MUST use classic (unencrypted) DNS instead of encryption. Similarly, if none of the SVCB records for the authoritative servers in the cache have information about encrypted services as described in [I-D.schwartz-svcb-dns], the resolver MUST use classic (unencrypted) DNS instead of encryption.

If there are any SVCB records in the cache for the authoritative servers for a zone with a positive response, the resolver MUST try each indicated authoritative server using DNS with encryption until it successfully sets up a connection. The resolver only attempts to use the encrypted transports that are in the associated SVCB record for the authoritative server. Reasons for TLS failures are listed in Section 3.1.

After a DNS with encryption session is set up, the resolver uses that authoritative server for whatever query about the zone it was going to send. If a resolver cannot set up a DNS with encryption session with any of the authoritative servers, it MUST attempt to perform the resolution over classic (unencrypted) DNS as it would have without encryption.

A resolver SHOULD keep a DNS with encryption session to a particular server open if it expects to send additional queries to that server in a short period of time. If the server closes the DNS with encryption session, the resolver can possibly re-establish a DNS with encryption session using encrypted session resumption. [RFC7766] says "both clients and servers SHOULD support connection reuse" for TCP connections, and that advice could apply as well for DNS with encryption even though DNS with encryption has greater overhead for saving state.

Privacy-oriented resolvers (defined in [RFC8932]) following this protocol MUST NOT indicate that they are using encryption because this protocol is susceptible to on-path attacks.

3.1. Resolver Session Failures

The following are the reasons that a DNS with encryption session might fail to be set up:

* The resolver receives a TCP RST response

* The resolver does not receive replies to TCP or TLS setup (such as getting the TCP SYN message, the first TLS message, or completing TLS handshakes)

* The TLS handshake gets a definitive failure

* The encrypted session fails for reasons other than for authentication, such as incorrect algorithm choices or TLS record failures

3.2. Resolver Process as Pseudocode

This section is meant as an informal clarification of the protocol, and is not normative. The pseudocode here is designed to show the intent of the protocol, so it is not optimized for things like intersection of sets and other shortcuts.

```
# Inputs
```

```
ns_names = List of NS Rdatas from the NS RRset for the queried name
can_do_secure = List of secure transports supported by resolver
secure_names_and_transports = Empty list, filled in below

# Does this resolver support any secure transports?
if length of can_do_secure is 0:
  query using classic DNS on any/all ns_names; finished

# Fill secure_names_and_transports with (name, transport) tuples
for this_name in ns_names:
  if signal_rrset(this_name) is in the resolver cache:
    if signal_rrset(this_name) is NXDOMAIN:
      continue
    for this_transport in signal_rrset(this_name):
      if this_transport in can_do_secure:
        add (this_name, this_transport) to secure_names_and_transports
  else: # if signal_rrset(this_name) is not in the resolver cache
    queue a query for signal_rrset(this_name) for later caching

# Query over secure transport until successful
for (this_name, this_transport) tuple in secure_names_and_transports:
  query using this_transport on this_name
  if successful:
    finished

# Got here if no this_name/this_transport query was successful
#   or if secure_names_and_transports was empty
query using classic DNS on any/all ns_names; finished
```

4. Serving with Encryption

An operator of an authoritative server following this protocol SHOULD publish SVCB records as described in Section 2. If they cannot publish such records, the security properties of their authoritative servers will not be found. If an operator wants to test serving using encryption, they can publish SVCB records with short TTLs and then stop serving with encryption after removing the SVCB records and waiting for the TTLs to expire.

An operator of authoritative servers for a zone that is following

this protocol MAY support encryption towards any IP address on which it offers service for classic DNS on port 53. It is acceptable for such an operator to only offer encryption on some of the named authoritative servers, such as when the operator is determining how far to roll out encrypted service.

A server MAY close an encrypted connection at any time. For example, it can close the session if it has not received a DNS query in a defined length of time. The server MAY close an encrypted session after it sends a DNS response; however, it might also want to keep the session open waiting for another DNS query from the resolver. [RFC7766] says "both clients and servers SHOULD support connection reuse" for TCP connections, and that advice could apply as well for DNS with encryption even though DNS with encryption has greater overhead for saving state.

5. Resolvers Reporting Errors to Authoritative Servers

Resolvers should have a method of telling authoritative servers that there are problems with the encrypted service they are offering. There is a proposal that the DNSOP Working Group might adopt [I-D.arends-dns-error-reporting], which would enable such reporting.

((Clearly, more will need to go here.))

6. IANA Considerations

((Update registration for TCP/853 to also include ADoT))

((Maybe other updates for DoH and DoQ))

7. Security Considerations

The method described in this document explicitly allows a resolver to perform DNS communications over traditional unencrypted, unauthenticated DNS on port 53, if it cannot find an authoritative server that advertises that it supports encryption. The method described in this document explicitly allows a resolver using encryption to choose to allow unauthenticated encryption. In either of these cases, the resulting communication will be susceptible to obvious and well-understood attacks from an attacker in the path of the communications.

An authoritative server that wants to only serve data to resolvers that using fully-authenticated encryption as described in [I-D.rescorla-dprive-adox-latest] cannot differentiate between those resolvers and resolvers using the mechanisms described in this document.

8. Acknowledgements

Puneet Sood contributed many ideas to early drafts of this document.

The DPRIVE Working Group has contributed many ideas that keep shifting the focus and content of this document.

9. References

9.1. Normative References

[I-D.ietf-dnsop-rfc8499bis]
Hoffman, P. and K. Fujiwara, "DNS Terminology", Work in Progress, Internet-Draft, draft-ietf-dnsop-rfc8499bis-01, 20 November 2020, <https://www.ietf.org/archive/id/draft-ietf-dnsop-rfc8499bis-01.txt>.

[I-D.ietf-dnsop-svcb-https]
Schwartz, B., 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-04, 17 March 2021, <https://www.ietf.org/archive/id/draft-ietf-dnsop-svcb-https-04.txt>.

[I-D.rescorla-dprive-adox-latest]
Pauly, T., Rescorla, E., Schinazi, D., and C. A. Wood, "Signaling Authoritative DNS Encryption", Work in Progress, Internet-Draft, draft-rescorla-dprive-adox-latest-00, 26 February 2021, <https://www.ietf.org/archive/id/draft-rescorla-dprive-adox-latest-00.txt>.

[I-D.schwartz-svcb-dns]
Schwartz, B., "Service Binding Mapping for DNS Servers", Work in Progress, Internet-Draft, draft-schwartz-svcb-dns-02, 17 February 2021, <https://www.ietf.org/archive/id/draft-schwartz-svcb-dns-02.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>.

[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>.

[RFC7766] Dickinson, J., Dickinson, S., Bellis, R., Mankin, A., and D. Wessels, "DNS Transport over TCP - Implementation Requirements", RFC 7766, DOI 10.17487/RFC7766, March 2016, <https://www.rfc-editor.org/info/rfc7766>.

[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>.

9.2. Informative References

[I-D.arends-dns-error-reporting]
Arends, R. and M. Larson, "DNS Error Reporting", Work in Progress, Internet-Draft, draft-arends-dns-error-reporting-00, 30 October 2020, <https://www.ietf.org/archive/id/draft-arends-dns-error-reporting-00.txt>.

[I-D.ietf-dprive-dnsoquic]
Huitema, C., Mankin, A., and S. Dickinson, "Specification of DNS over Dedicated QUIC Connections", Work in Progress,

Internet-Draft, draft-ietf-dprive-dnsoquic-02, 22 February 2021, <https://www.ietf.org/archive/id/draft-ietf-dprive-dnsoquic-02.txt>.

[RFC8932] Dickinson, S., Overeinder, B., van Rijswijk-Deij, R., and A. Mankin, "Recommendations for DNS Privacy Service Operators", BCP 232, RFC 8932, DOI 10.17487/RFC8932, October 2020, <https://www.rfc-editor.org/info/rfc8932>.

[RSO_STATEMENT]
"Statement on DNS Encryption", 2021, <https://root-servers.org/media/news/Statement_on_DNS_Encryption.pdf>.

Authors' Addresses

Paul Hoffman
ICANN

Email: paul.hoffman@icann.org

Peter van Dijk
PowerDNS

Email: peter.van.dijk@powerdns.com