Workgroup: dprive
Internet-Draft:
draft-schwartz-dprive-name-signal-00
Published: 8 June 2021
Intended Status: Experimental
Expires: 10 December 2021
Authors: B. Schwartz W. Kumari
Google LLC Google LLC

Nameserver Access Modes with Encryption Held in Alphanumeric Configuration Keys

Abstract

Some recent proposals to the DPRIVE working group rely on the use of SVCB records to provide instructions about how to reach an authoritative nameserver over an encrypted transport. These proposals will be difficult to deploy until the parent domain's delegation software has been modified to support these records. As an interim solution for these domains, this draft proposes encoding relevant signals in the child's NS-name.

Discussion Venues

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

Discussion of this document takes place on the mailing list (dns-privacy@ietf.org), which is archived at https://mailarchive.ietf.org/arch/browse/dns-privacy/.

Source for this draft and an issue tracker can be found at https://github.com/wkumari/draft-schwartz-dprive-name-signal.

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 10 December 2021.

Table of Contents

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. Background

[I-D.draft-schwartz-svcb-dns] defines how to use SVCB records to describe the secure transport protocols supported by a DNS server.

[I-D.draft-ietf-dprive-unauth-to-authoritative] describes the use of such records on the names of nameservers (the "NS name") to enable opportunistic encryption of recursive-to-authoritative DNS queries. Resolvers are permitted to fetch SVCB records asynchronously and cache them, resulting in "partial opportunistic encryption": even without an active adversary forcing a downgrade, queries will sometimes be sent in cleartext. Participating authoritative nameservers and recursive resolvers would have to be modified to make use of these records.

When the child zone is DNSSEC-signed, publishing a SVCB record of this kind is technically sufficient to enable authenticated encryption. However, in order to support reliable authentication, recursive resolvers would have to query for a SVCB record on every signed delegation, and wait for a response before issuing their intended query. We call this behavior a "synchronous binding check".

Many validating resolvers might not be willing to enable a "synchronous binding check" behavior, as this would slow down resolution of many existing domains in order to enable a new feature (authenticated encryption) that is not yet used at all. To enable authenticated encryption without this general performance loss, [I-D.draft-rescorla-dprive-adox-latest] proposes to deliver the SVCB records from the parent, in the delegation response. This avoids the need for a binding check, at the cost of additionally requiring modifications to the parent nameserver, which must provide these extra records in delegation responses.

Providing these additional records is sufficient to enable "full opportunistic encryption": the transport is always encrypted in the absence of an active adversary. However, these records are not protected by DNSSEC, so the child can only achieve fully authenticated encryption if the parent also implements fully authenticated encryption or otherwise protects the delivery of these records.

Even if this approach is standardized, many parent zones may not support delivery of SVCB records in delegation responses in the near future. To enable the broadest use of encrypted transport, we may need an interim solution that can be deployed more easily.

3. Proposal

We propose to indicate a nameserver's support for encrypted transports using a signal encoded in its name. This signal takes two forms: a "flag" and a "menu".

QUESTION: Do we need both of these forms, or should we drop one?

We note that encoding semantics in DNS labels is a hack, but believe that the privacy benefits outweigh the ick factor.

In either form, the signal helps resolvers to acquire a SVCB RRSet for the nameserver. Resolvers use this RRSet as specified in [I-D.draft-rescorla-dprive-adox-latest].

3.1. Flag form

If the NS name's first label is svcb, this is regarded as a "flag". When contacting a flagged nameserver, participating resolvers SHOULD perform a synchronous binding check, and upgrade to a secure transport if appropriate, before issuing the query.

The presence of this flag does not guarantee that the corresponding SVCB records are actually present.

3.2. Menu form

If the NS name's first label starts with svcb--, the label's subsequent characters represent a "menu" of connection options, which can be decoded into a SVCB RRSet. To decode the RRSet, each character is transformed into a SVCB RR with the following components:

- The owner name is the NS name plus the prefix label "_dns".
- The SvcPriority is the character's order in the list (starting at 1)
- The TargetName is the NS name
- The SvcParams are indicated in the registry entry for this menu character (Section 6).

For example, the name "svcb-qt.ns3.example." would be decoded to this RRSet:

```
_dns.svcb--qt.ns3.example. IN SVCB 1 svcb--qt.ns3.example. alpn=doq
_dns.svcb--qt.ns3.example. IN SVCB 2 svcb--qt.ns3.example. alpn=dot
```

The menu characters are a-z and 0-9; all other characters are reserved for future use. Upon encountering any character outside

these ranges, parsers MUST stop and return successfully. Parsers MUST ignore characters that are allowed but not recognized.

> QUESTION: Do we need more than 36 codepoints? Is there a nice simple format that would give us a lot more codepoints?
>
> QUESTION: Should we consider a format that actually encodes the SvcParams in the label instead?

3.3. Implementation requirements

Resolvers that implement support for "menu" mode MUST also support the "flag" mode. Resolvers that support either mode MUST also support [I-D.draft-rescorla-dprive-adox-latest], and ignore the in-name signal if any SVCB records are included in a delegation response.

When possible, zones SHOULD use SVCB records in the delegation response and omit any in-name signal.

4. Security Considerations

NS names received during delegation are not protected by DNSSEC. Therefore, just like in [I-D.draft-rescorla-dprive-adox-latest], this scheme only enables authenticated encryption if the parent domain can provide authentication without DNSSEC validation, e.g. using a secure transport or Zone Digest [RFC8976].

> QUESTION: Do we expect to have parent zones that can provide authenticated NS names but cannot provide authenticated SVCB records in delegation responses? (Maybe the root, with ZONEMD?) If not, does this proposal provide enough value?

5. Operational Considerations

It is possible that an existing NS name already matches the "flag" pattern. Such a "false positive flag" will result in a small performance loss due to the unnecessary synchronous binding check, but will not otherwise impair functionality.

If a pre-existing NS name contains the menu pattern, that nameserver will become unreachable by resolvers implementing this specification. The authors believe that no such nameservers are currently deployed, and such servers are unlikely to be deployed by accident.

6. IANA Considerations

IANA is requested to create a new registry entitled "Authoritative Server Transport In-Name Signal Characters", with the following fields:

*Character: a digit or lower-case letter

*SvcParams: a valid SVCB SvcParams set in presentation format

The registry policy is **TBD**.

The initial contents (**DO NOT USE**, subject to change) are as follows:

Character	SvcParams
t	alpn=dot
h	alpn=h2 dohpath=/dns-query{?dns}
3	alpn=h3 dohpath=/dns-query{?dns}
q	alpn=doq

Table 1

7. References

7.1. Normative References

[I-D.draft-schwartz-svcb-dns]
Schwartz, B., "Service Binding Mapping for DNS Servers", Work in Progress, Internet-Draft, draft-schwartz-svcb-dns-03, 19 April 2021, <https://www.ietf.org/archive/id/draft-schwartz-svcb-dns-03.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>.

[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>.

7.2. Informative References

[I-D.draft-bretelle-dprive-dot-spki-in-ns-name]
Bretelle, E., "Encoding DNS-over-TLS (DoT) Subject Public Key Info (SPKI) in Name Server name", Work in Progress, Internet-Draft, draft-bretelle-dprive-dot-spki-in-ns-

name-00, 11 March 2019, <https://www.ietf.org/archive/id/draft-bretelle-dprive-dot-spki-in-ns-name-00.txt>.

[I-D.draft-fujiwara-dnsop-delegation-information-signer]
Fujiwara, K., "Delegation Information (Referrals) Signer for DNSSEC", Work in Progress, Internet-Draft, draft-fujiwara-dnsop-delegation-information-signer-00, 2 November 2020, <https://www.ietf.org/archive/id/draft-fujiwara-dnsop-delegation-information-signer-00.txt>.

[I-D.draft-ietf-dprive-unauth-to-authoritative] Hoffman, P. and P. V. Dijk, "Recursive to Authoritative DNS with Unauthenticated Encryption", Work in Progress, Internet-Draft, draft-ietf-dprive-unauth-to-authoritative-01, 19 May 2021, <https://www.ietf.org/archive/id/draft-ietf-dprive-unauth-to-authoritative-01.txt>.

[I-D.draft-levine-dprive-signal-02]
Levine, J., "Signaling That an Authoritative DNS server offers DoT", Work in Progress, Internet-Draft, draft-levine-dprive-signal-02, 17 November 2019, <http://www.ietf.org/internet-drafts/draft-levine-dprive-signal-02.txt>.

[I-D.draft-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.draft-vandijk-dnsop-ds-digest-verbatim]
Dijk, P. V., "The VERBATIM Digest Algorithm for DS records", Work in Progress, Internet-Draft, draft-vandijk-dnsop-ds-digest-verbatim-00, 25 September 2020, <https://www.ietf.org/archive/id/draft-vandijk-dnsop-ds-digest-verbatim-00.txt>.

[I-D.draft-vandijk-dprive-ds-dot-signal-and-pin] Dijk, P. V., Geuze, R., and E. Bretelle, "Signalling Authoritative DoT support in DS records, with key pinning", Work in Progress, Internet-Draft, draft-vandijk-dprive-ds-dot-signal-and-pin-01, 13 July 2020, <https://www.ietf.org/archive/id/draft-vandijk-dprive-ds-dot-signal-and-pin-01.txt>.

[RFC8976] Wessels, D., Barber, P., Weinberg, M., Kumari, W., and W. Hardaker, "Message Digest for DNS Zones", RFC 8976, DOI

10.17487/RFC8976, February 2021, <https://www.rfc-editor.org/info/rfc8976>.

Appendix A. Comparison with related designs

Several other designs have been proposed to encode a transport upgrade signal in an existing record type.

A.1. Indicating DoT support with a name prefix

Section 3.6 of [I-D.draft-levine-dprive-signal-02] discusses using the "xs-" name prefix to indicate support for DNS over TLS. This is equivalent to a "svcb-t" label in this formulation. This draft may be seen as an expansion of that proposal, harmonized with the SVCB-based discovery drafts.

A.2. Encoding the SPKI pin in the leaf label

[I-D.draft-bretelle-dprive-dot-spki-in-ns-name] also proposes to encode a signal in the leaf label. The signal includes an SPKI pin, for authentication of the TLS connection.

Including an SPKI pin allows authentication of the nameserver without relying on DANE or PKI validation. However, like this draft, it does not achieve authenticated encryption unless the NS name can be delivered securely during delegation. It may also create operational challenges when rotating TLS keys, due to the need to update the parent zone.

A.3. Encoding the signal in an additional NS record

It would be possible to encode the signal by adding a special NS record to the RRSet. This would avoid the need to rename any existing nameservers. However, this arrangement has different semantics: it is scoped to the entire child zone, rather than a specific nameserver. It also relies heavily on existing resolvers having robust and performant fallback behavior, which may not be a safe assumption.

(Credit: Paul Hoffman)

A.4. Extending the DS record

[I-D.draft-vandijk-dprive-ds-dot-signal-and-pin] encodes a signal and pin in a DS record by allocating a new fake "signature algorithm" and encoding the TLS SPKI in a DNSKEY record. This enables fully authenticated encryption (only requiring that the parent zone is signed). However, it has very limited flexibility for representing different transport configurations, and creates challenges during TLS key rotation.

A.5. Enabling authentication of delegation data

[I-D.draft-fujiwara-dnsop-delegation-information-signer] adds a DS record over the delegation information. When combined with this draft, this would enable fully authenticated encrypted transport. However, this approach requires very tight coherence between the child and parent (e.g. when removing a nameserver) that may not be achievable in practice.

[I-D.draft-vandijk-dnsop-ds-digest-verbatim] allows children to push arbitrary authenticated delegation data into the parent. This could be used to convey SVCB RRSets for the delegation securely. However, it requires parents to accept a new digest type, and bends the usual DS semantics even further.

Acknowledgments

TODO

Authors' Addresses

Benjamin M. Schwartz
Google LLC

Email: bemasc@google.com

Warren Kumari
Google LLC

Email: warren@kumari.net