Workgroup: HTTP
Internet-Draft:
draft-thomson-httpbis-alt-svcb-01
Obsoletes: 7838 (if approved)
Published: 31 March 2023
Intended Status: Standards Track
Expires: 2 October 2023
Authors: M. Thomson
Mozilla
M. Bishop
Akamai Technologies
L. Pardue
Cloudflare
T. Jensen
Microsoft

HTTP Alternative Services, Plan B

Abstract

HTTP servers deployments that include multiple service endpoints can use alternative services to direct clients to use a different service endpoint.

This document obsoletes RFC 7838.

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://martinthomson.github.io/alt-svcb/draft-thomson-httpbis-alt-svcb.html. Status information for this document may be found at https://datatracker.ietf.org/doc/draft-thomson-httpbis-alt-svcb/.

Discussion of this document takes place on the HTTP Working Group mailing list (mailto:ietf-http-wg@w3.org), which is archived at https://lists.w3.org/Archives/Public/ietf-http-wg/.

Source for this draft and an issue tracker can be found at https://github.com/martinthomson/alt-svcb.

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 2 October 2023.

Copyright Notice

Table of Contents

1. Introduction

HTTP origins are often comprised of multiple service endpoints. This can be driven by multiple requirements, such as a need to scale by adding multiple physical servers, the need to place endpoints in network locations that are closer to clients for performance reasons, or the need to support multiple HTTP versions, like HTTP/2 [HTTP/2] or HTTP/3 [HTTP/3].

For servers that operate multiple service endpoints, it can be advantageous to have clients make requests to a specific service endpoint.

* Some deployments might seek to direct a client to a service endpoint that is better able to serve requests for that client. This might occur if DNS resolution of the server name produces the address of a server instance that is further from the client.

* Servers might seek to reduce load, perhaps in anticipation of an imminent shutdown or maintenance action. An alternative service declaration can reduce either server load or the number of clients that might be affected.

* Many deployments of HTTP/3 [HTTP/3] use the protocol identifiers in an alternative service declaration to make clients aware of support for the newer protocol.

HTTP alternative services provide a means of indicating to clients which service endpoints a server would prefer be used for future requests. Clients use alternative service advertisements as prompt to discover and use these more preferred service endpoints.

Clients that learn about an alternative service can establish a connection to the identified service endpoint, which - if successfully established and authenticated - is then used for future requests. Any existing connections the client has are retained and used until the new connection is successful. This ensures that clients can continue making requests of the server without interruption.

1.1. Previous Alternative Services Designs

RFC 7838 [ALT-SVC] provided the first alternative service design for HTTP. This design turned out to have a number of shortcomings in deployment. Though these issues were anticipated in the design, the measures that were used often did not work particularly well.

The RFC 7838 design included caching logic based on setting an "ma" (or max-age) parameter. This turned out to be challenging for many server deployments. Setting too large a max-age meant that clients used the indicated service endpoint for longer than was desired when operating conditions changed. Conversely, a short cache period for an advertisement for HTTP/3 resulted in frequently reverting to previous versions on subsequent connections.

Alternative services turned out to interact poorly with service configuration information that is published in the DNS. With the introduction of HTTPS records [SVCB], more details of service endpoints can be advertised in the DNS, including the support for HTTP/3. But this created two independent sources of this information, each with its own approach to caching.

Alternative services are dependent on networking conditions. RFC 7838 attempted to manage this by having clients be responsible for invalidating alternatives when changes in their network are detected, unless the alternative is explicitly marked as "persistent". In practice, detecting the necessary changes is difficult for many clients, so this requirement is not consistently implemented.

The result being that the alternative services mechanisms defined in RFC 7838 produced suboptimal or even detrimental outcomes in some deployments.

This document obsoletes RFC 7838.

1.2. A New Alternative

This document describes a different approach to advertising alternative services. This approach uses the DNS as the singular source of information about service reachability. An alternative service advertisement only acts as a prompt for clients to seek updated information from the DNS.

To use this new design, a server advertises an alternative name using the "Alt-SvcB" field.

```
200 OK HTTP/1.1
date: Mon, 24 Oct 2022 02:58:31 GMT
alt-svcb: "instance31.example.com"
content-length: 0
```

Clients can then consult the DNS, making HTTPS queries [SVCB] starting with this name. The alternative name is used in place of the name of the authority and using HTTPS records is mandatory, but the process otherwise follows normal HTTPS record resolution and connection procedures. Section 2 defines how this name is used in detail.

Future connections for requests to resources on the same server use HTTPS record resolution to the name of the authority, but are reprioritized if a successful connection was previously made to an alternative service. Section 2.2 defines how this process works in more detail.

1.3. 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.

The terms "server" and "client" are defined in [HTTP]. The term "origin" is defined in [ORIGIN].

The term "alternative name" refers to the name advertised by a server to a client. This refers to a domain name that is queried by the client to discover both service names and service endpoints.

A "service name" is the TargetName from an HTTPS ServiceMode record [SVCB]. Service names, their associated parameters (SvcParams), and IP addresses describe a "service endpoint". Clients estabiliish connections to service endpoints in order to make requests of a server.

A server is identified using its "origin name", which is the domain name from the target URI of resources the client makes requests toward. This is the name that the client authenticates when determining if a service endpoint is authoritative. Unlike an alternative name or service name, an origin name can be an IP address rather than a domain name.

There can be different values for origin name, alternative name, and service name.

2. Using Alternative Services

A server advertises the availability of alternative services by providing the client with an alternative name. The server does this using either a field in a response (Section 3.1) or an HTTP/2 or HTTP/3 frame (Section 3.2).

When a client receives a new alternative name from a server, they **SHOULD** attempt to discover and use the service endpoints referred to by that name for future requests to that server.

In order to discover and use the identified service endpoints, the client attempts to make a request for a resource on the same server using the provided alternative name as follows:

1. The client makes a DNS query for HTTPS records for the alternative name, following the procedures in Section 3 of [SVCB]. Clients make this query as a "SVCB-reliant" client, treating missing or unobtainable HTTPS records as a failure. If this process fails to produce service parameters or IP addresses, the process is aborted.

2. The client establishes a connection using the service parameters and addresses learned from the DNS query. The client uses the origin name in any TLS server name indication [SNI] of the server name from the URL, not the alternative name. This allows the server to produce a certificate for the origin name, which the client can validate as applying to the URL it is resolving. If a connection cannot be established, the process is aborted.

3. The client validates that the server is authoritative for the resource using the server origin name. If the server is not authoritative, the process is aborted.

4. The client makes a query for the resource. If the server does not respond or responds with a 421 (Misdirected Request) (see Section 15.5.20 of [HTTP]), the process is aborted. A client **MAY** re-attempt a request or request another resource if the server responds with a 5xx status code (see Section 15.6 of [HTTP]).

5. Once a response is received, the connection to the alternative service endpoint is complete. Any other connections can be closed and future requests directed to the new connection. The client **SHOULD** remember the alternative name and the service name that were used; see Section 2.1.

A client **MAY** send multiple requests using the newly established connection to the alternative service after it verifies that the

server is authoritative. However, a client **MUST NOT** remember a service name until at least one request has been successfully completed with a 2xx or 3xx status code. The alternative service is therefore active once the connection is established, but it will not be reused (Section 2.2) for future connections until a request completes successfully.

A client **MAY** continue sending other requests over any existing connection to the server until this process completes in order to minimize latency for those requests. A client **MAY** - when presented with an alternative name - proactively make a request for an arbitrary resource on the server, rather than waiting for the next time a request is needed. This might allow the connection to be available for future requests with less delay.

2.1. Retention of Alternatives

Clients **SHOULD** remember the successful use of an alternative service in order to support reuse (Section 2.2). Two pieces of information are retained:

* the alternative name, which is the name provided by the server in the Alt-SvcB field or ALTSVCB frame, and

* the service name, which is the TargetName from the ServiceMode HTTPS record that was used to successfully connect to the server.

These two names are saved for the server against the origin of the server [ORIGIN]. Clients **MUST NOT** reuse saved information for a server with a different hostname, port, or scheme.

The alternative name, as carried in an Alt-SvcB field or ALTSVCB frame, is retained only so that the client can avoid repeated attempts to discover and connect to alternative services. A server can send Alt-SvcB fields in multiple responses or send multiple ALTSVCB frames. Repeating the discovery process could be wasteful for a client.

Any time that a server provides a different name in an Alt-SvcB field or ALTSVCB frame, any existing information **MUST** be discarded. A client **MAY** then initiate a DNS query and connection attempt using the new alternative name.

Though a server might repeat an alternative name, clients **MUST NOT** consider the absence of an Alt-SvcB field in a response as indicative of a retraction of a previous advertisement. An alternative name is only removed when replaced with a different alternative name or when a remembered service name does not appear in the set of HTTPS query responses (see Section 2.2).

After a failed attempt to use an alternative service, a failure is remembered by retaining the alternative name without a service name. This avoids making repeated attempts to use an alternative service that is not available, even if the server repeats the alternative name. A client **MAY** periodically attempt to retry a failed alternative if the information is repeated.

A server can explicitly request that a client remove any remembered service name by providing an alternative name of "invalid". The "invalid" domain name corresponds to a DNS name that will never successfully resolve (see Section 6.4 of [SUDN]), which guarantees that an attempt to use this name cannot succeed. Clients **MAY** recognize the alternative name "invalid" as special and avoid any attempt to use this to discover an alternative service.

2.2. Reusing Alternatives

In subsequent connections to the same origin, clients make a DNS query for HTTPS records for the origin name. If, after following any CNAME or AliasMode records, this query returns a ServiceMode resource record (RR) that includes a TargetName that is identical to the service name that is remembered for the request origin, the client **SHOULD** choose that over any alternatives. This ignores any SvcPriority attributes that might cause other records to be chosen and includes any RRs that are marked "alt-only"; see Section 2.2.2.

Note that when reusing an alternative service, a client does not make a query for the remembered alternative name. HTTPS queries are made for the origin name, which is the domain name from the target URI of the request; see also Section 2.3.

If a query for HTTPS records does not produce a ServiceMode record with a TargetName that matches the remembered service name, all remembered information **MUST** be removed for that origin. The client then uses the normal SVCB-optional resolution logic as defined in [SVCB].

When reusing stored information, if a connection attempt is unsuccessful (see Section 2), remembered information for that origin **MUST** be removed. Clients clear retained alternative service information on reuse to prevent stale information from affecting all future connection attempts. After removing remembered information, a client **MAY** make another attempt to connect using any other ServiceMode records that the DNS query produced.

2.2.1. Example of Reuse

A client that is fetching "https://example.com/" might originally perform a DNS query for "example.com" and receive in response:

```
example.com. 7200 IN HTTPS 1 . port=443
example.com. 7200 IN HTTPS 10 alt1.example. port=8443
example.com. 7200 IN HTTPS 10 alt2.example. port=8443
example.com. 7200 IN HTTPS 10 alt2.example. port=8443
```

Under normal conditions, the SvcPriority of the "alt?.example" RRs would indicate that they are not preferred, so the "example.com" record would be used.

If the client received an alternative service advertisement from this server for "alt.example.net" it would then make a DNS query to that name. This might return a different set of records, as follows:

```
alt.example.net. 7200 IN HTTPS 1 alt2.example. port=8887 alpn=h3
alt.example.net. 7200 IN HTTPS 1 alt3.example. port=8887 alpn=h3
```

If the client selects "alt2.example" and successfully connects to that host, it remembers both the alternative name ("alt.example.net") and a service name ("alt2.example").

In subsequent connections to "example.com", the client again queries the "example.com" name. Importantly, this is the origin name and not any other name it might have remembered. The resulting response - after following indirections through AliasMode, CNAME, or similar mechanisms - produces the same records as previously (perhaps because these were retained in a cache):

```
example.com. 7200 IN HTTPS 1 . port=443
example.com. 7200 IN HTTPS 10 alt1.example. port=8443
example.com. 7200 IN HTTPS 10 alt2.example. port=8443
example.com. 7200 IN HTTPS 10 alt2.example. port=8443
```

The ServiceMode HTTPS record for "alt2.example" is used, even though this is a lower priority than other records. It is also used despite not using the same port number or protocol as the previous successful connection.

2.2.2. Exclusive Alternative Services

ServiceMode HTTPS records can be marked as only being available for use as an alternative. This allows servers to use alternative services for specific server instances, without having clients connect to them without being first invited to do so.

This is achieved with a SvcParam with a key of "alt-only" (codepoint TBD). The value of this SvcParamKey **MUST** be empty. HTTPS ServiceMode records with this SvcParamKey **MUST NOT** be used unless the client is actively seeking an alternative, either as a result of actively looking up an alternative name or because the alternative has been remembered.

To prevent clients that do not support this specification from using these services, the "alt-only" SvcParamKey **MUST** be listed in the "mandatory" SvcParam.

In the following example, though "alt1.example" is listed at a higher priority than "example.com", clients will not use this service unless an alternative was provided by the server:

```
example.com. 7200 IN HTTPS 1 alt1.example. port=443 alt-only mandatory=a
example.com.  7200 IN HTTPS 2 . port=443
```

2.3. Servers Identified by IP

An alternative name can be provided by a server that is identified by an IP address or host names that are not domain names. However, HTTPS queries cannot be made for servers that are not identified by a domain name. This makes it impossible to use such identifiers. A client **MAY** disable alternative services for servers that are not identified by a domain name.

2.4. Port Numbers

An alternative name provided in an Alt-SvcB field or ALTSVCB frame can be any valid DNS QNAME. This includes those with underscored labels [ATTRLEAF] and those that might be used to query for HTTPS records to a non-default port.

```
200 OK HTTP/1.1
date: Mon, 24 Oct 2022 02:58:31 GMT
alt-svcb: "_8443._https.example.com"
content-length: 0
```

This might be used to direct clients to connect to alternative ports using existing records. Note that the HTTPS records might direct clients to an entirely different port number than the name implies. Clients **MUST NOT** infer a port number from the provided name, treating this name no differently than any other and using the port number derived from the service parameters.

2.5. Interaction with GOAWAY

Servers that advertise alternative services cannot expect clients to switch to the advertised alternative. Use of any alternative is entirely at the discretion of clients. If the client is unsuccessful in connecting to an alternative or does not attempt a connection, they could continue to use the existing connection for new requests.

A server that seeks to actively encourage clients to disconnect and seek service elsewhere needs to use graceful shutdown procedures of

the HTTP version that is in use. HTTP/2 [HTTP/2] and HTTP/3 [HTTP/3] each provide a GOAWAY frame that can be used to initiate the graceful shutdown of a connection. Alternative services is not a substitute for these mechanisms.

2.6. Proxies

The procedures in this document apply to clients that connect to gateways or reverse proxies. However, clients that connect via a proxy, using HTTP CONNECT or similar methods, have a choice.

Clients that provide a proxy with the origin name of a server leave name resolution to the proxy. Such a client **MUST** ignore any alternative service advertisement it receives. These clients **MAY** fallback to using legacy alternative services; see Section 2.7.

Clients that make HTTPS queries for any connection attempt via a proxy can use alternative services. Such a client can provide the proxy with the IP address of the server it wishes to contact, rather than providing a name.

2.7. Fallback to Alt-Svc

A client that successfully makes use of HTTPS records in resolving an origin name or alternative name **MUST** ignore any Alt-Svc fields or ALTSVC frames [ALT-SVC] that the server provides. This document obsoletes the mechanisms defined in RFC 7838 [ALT-SVC].

Servers might provide Alt-Svc fields or ALTSVC frames [ALT-SVC] in order to support clients that cannot use HTTPS records.

2.8. Authority For Service Endpoint Configuration

This design does not assume that information provided by a server or by the DNS is authoritative information about the configuration of service endpoints. This is despite the information in Alt-SvcB fields or ALTSVCB frames being provided by a server that is authoritative.

Instead, once a server is determined to be authorative (see Section 4.3 of [HTTP]), that server is treated as the authority on all aspects of its own configuration. For example, with protocol selection, [ALPN] and maybe [SNIP] extensions in the TLS handshake [TLS] determine what protocol is used.

For requests, a server that is determined to be authoritative for an origin can answer all requests on that origin. All service endpoints that are authoritative **SHOULD** provide equivalent service to any other, though they could differ in terms of performance, diagnostic information, or other minor details. Clients will expect

service endpoints to provide equivalent - or perhaps identical - service.

3. Protocol Elements

Multiple ways of advertising alternative services are defined. The Alt-SvcB field in Section 3.1 allows servers to indicate a preferred service in responses. The ALTSVCB frames in Section 3.2 allows a server to provide alternative names outside of the context of a query.

These approaches have different properties. Alt-SvcB fields are forwarded by intermediaries and so might reach clients through a gateway or reverse proxy. Clients that use a proxy without using CONNECT or similar tunnels, might also receive an alternative name using a field. In comparison, ALTSVCB frames each only apply to a single origin within the scope of a single connection.

3.1. Alt-SvcB Field

The "Alt-SvcB" response field is a List of String values (see Sections 3.1 and 3.3.3 of [STRUCTURED-FIELDS]). This response field **MAY** appear in a header or trailer section, though servers need to be aware that some clients might not process field values.

Each field value includes an alternative name. Each alternative name is encoded as an ASCII string, or a series of DNS A-labels, each separated by a single period character (".", U+2E). Each value **MAY** end with a period, though - for the purposes of the process in Section 2 - the string is treated as an absolute DNS QNAME whether or not a trailing period is present.

The applicable origin is derived from the origin of the target URI; see Section 7.1 of [HTTP] and [ORIGIN].

If multiple Alt-SvcB fields or field values are present in a response, the client **MAY** use any subset of the provided alternative names, including none, one, or all of the provided names.

Servers **SHOULD NOT** provide more than one name. The DNS provides ample opportunity to present clients with multiple options, including the use of priority to help manage selection. A list is tolerated only to allow for the possibility that multiple field lines might be added to responses without proper coordination.

Clients **MUST** ignore unknown parameters that are provided with alternative names. This document does not define any parameters as the DNS is expected to provide supplementary information about services; a revision of this document would be required to enable the use of parameters.

3.2. ALTSVCB Frame

An ALTSVCB frame is defined for both HTTP/2 and HTTP/3. The frame provides an alternative name for an identified origin [ORIGIN].

In both protocols, the ALTSVCB frame uses the identifier TBD. The format for both protocols is the same; this is shown in Figure 1 using the notation from Section 3 of [QUIC].

```
ALTSVCB Frame {
  Origin Length (i),
  Origin (..),
  Alternative Name (..),
}
```

Figure 1: ALTSVCB Frame Format

The fields in the ALTSVCB frame are defined as follows:

Origin Length: An integer, encoded as a QUIC variable-length integer (see Section 16 of [QUIC]) indicating the length of the Origin field, in bytes.

Origin: The ASCII serialization of the affected origin; see Section 6.2 of [ORIGIN].

Alternative Name: The remainder of the frame contains a single alternative name, encoded as an ASCII string; see the definition in Section 3.1 for more details on the encoding.

If a server sends multiple ALTSVCB frames for the same origin, clients **MUST** ignore any frames other than the most recent.

4. Security Considerations

Alternative services present servers with a way of influencing how clients select service endpoints. This does not change how a service endpoint might be determined to be authoritative (even more so than its predecessor; see Appendix A).

4.1. Selecting Service Endpoints

This design assumes a Dolev-Yao attacker as is typical for Internet protocols [RFC3552]. This model assumes that an attacker has complete control of the network.

This design only supports HTTPS. Cleartext HTTP, such as might be used for URIs with a scheme of "http", is not supported. This means that TLS [TLS] is always used to establish whether a service

endpoint is authoritative, according to Section 4.3.3 of [HTTP]. TLS protects the configuration of service endpoints, including the choice of protocol; see Section 2.8. Furthermore, TLS prevents an attacker from inspecting or modifying the content of connections.

Even with TLS, a client connects to a service endpoint of the attacker's choice. This is a property of HTTP that the use of alternative services does not change, as the choice of service endpoint (including IP address and port number) is not authenticated when establishing a connection.

Certificates used to establish authority for HTTP servers do not include a port number, which means that all HTTP services that have a certificate for the same name will be treated by clients as being potentially authoritative. Section 4.3.3 of [HTTP] mandates checks on the target URI to mitigate this attack. Servers can use a 421 (Misdirected Request) status code (see Section 15.5.20 of [HTTP]) to signal any error and avoid the service endpoint being used.

DNS is not assumed to be secure in this threat model. The use of DNSSEC [DNSSEC] can ensure that clients do not receive incorrect information from DNS queries. However, DNSSEC does not defend against attacks on routing or forwarding infrastructure that might result in connections being directed toward a service endpoint chosen by an attacker. Using DNSSEC therefore does not change this analysis, though it can make attacks less feasible for some classes of attacker and so use is encouraged.

4.2. Attacks From Within Servers

In addition to network-based attackers, we also consider the possibility that an alternative service is advertised by an adversary who is able to generate HTTP responses. An adversary might be given the ability to generate responses for a subset of the resources on a server, where they might provide an Alt-SvcB field in a response.

This gives such an adversary some ability to direct clients toward a service endpoint of their choosing; see Section 4.1. It also potentially allows an adversary to create an unending sequence of alternatives; see Section 4.4.

Servers can mitigate these risks by restricting access to the ability of advertising an alternative name.

4.3. Tracking Clients

Remembering alternative names and service names might allow a server to connect activity at different times to the same client. Clients might be assigned a unique alternative name and service name in

order to make return connections identifiable. The need for the service name to appears the set of HTTPS records at the origin name does limit the ability of servers to track individual clients at scale, but this still might be used to separate clients into groups for tracking purposes or to track specific individuals.

Clients that clear origin-specific state in order to manage the risk of tracking **MUST** remove any remembered alternative service information when clearing state for a server (typically, this is associated with clearing cookies [RFC6265]).

4.4. Multiple Alternatives in Sequence

A client might receive multiple different alternative names in sequence, causing it to spend additional resources in discovering and connecting to different service endpoints. Repeatedly making connections can adversely affect performance.

This might be caused by a loop where the alternative name provided by each service endpoint points to the other or simply an unending sequence of new alternative names. This can arise if service endpoints are poorly configured.

A client can limit the effect of such misconfiguration by ignoring alternative names that change too frequently. A client might then continue to use the service endpoint to which it is connected or disable alternative services entirely for that origin.

5. Internationalization Considerations

An internationalized domain name that appears in either an Alt-SvcB field (Section 3.1) or an ALTSVCB frame (Section 3.2) **MUST** be expressed using A-labels; see Section 2.3.2.1 of [RFC5890].

6. IANA Considerations

TODO register:

*Field

*H2 Frame

*H3 Frame

*alt-only SvcParam

7. References

7.1. Normative References

[ALPN] Friedl, S., Popov, A., Langley, A., and E. Stephan, "Transport Layer Security (TLS) Application-Layer Protocol Negotiation Extension", RFC 7301, DOI 10.17487/RFC7301, July 2014, <https://www.rfc-editor.org/rfc/rfc7301>.

[HTTP] Fielding, R., Ed., Nottingham, M., Ed., and J. Reschke, Ed., "HTTP Semantics", STD 97, RFC 9110, DOI 10.17487/RFC9110, June 2022, <https://www.rfc-editor.org/rfc/rfc9110>.

[ORIGIN] Barth, A., "The Web Origin Concept", RFC 6454, DOI 10.17487/RFC6454, December 2011, <https://www.rfc-editor.org/rfc/rfc6454>.

[QUIC] Iyengar, J., Ed. and M. Thomson, Ed., "QUIC: A UDP-Based Multiplexed and Secure Transport", RFC 9000, DOI 10.17487/RFC9000, May 2021, <https://www.rfc-editor.org/rfc/rfc9000>.

[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>.

[RFC5890] Klensin, J., "Internationalized Domain Names for Applications (IDNA): Definitions and Document Framework", RFC 5890, DOI 10.17487/RFC5890, August 2010, <https://www.rfc-editor.org/rfc/rfc5890>.

[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>.

[SNI] Eastlake 3rd, D., "Transport Layer Security (TLS) Extensions: Extension Definitions", RFC 6066, DOI 10.17487/RFC6066, January 2011, <https://www.rfc-editor.org/rfc/rfc6066>.

[STRUCTURED-FIELDS] Nottingham, M. and P-H. Kamp, "Structured Field Values for HTTP", RFC 8941, DOI 10.17487/RFC8941, February 2021, <https://www.rfc-editor.org/rfc/rfc8941>.

[SVCB] 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>.

[TLS] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <https://www.rfc-editor.org/rfc/rfc8446>.

7.2. Informative References

[ALT-SVC] Nottingham, M., McManus, P., and J. Reschke, "HTTP Alternative Services", RFC 7838, DOI 10.17487/RFC7838, April 2016, <https://www.rfc-editor.org/rfc/rfc7838>.

[ATTRLEAF] Crocker, D., "Scoped Interpretation of DNS Resource Records through "Underscored" Naming of Attribute Leaves", BCP 222, RFC 8552, DOI 10.17487/RFC8552, March 2019, <https://www.rfc-editor.org/rfc/rfc8552>.

[DNSSEC] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "DNS Security Introduction and Requirements", RFC 4033, DOI 10.17487/RFC4033, March 2005, <https://www.rfc-editor.org/rfc/rfc4033>.

[HTTP/2] Thomson, M., Ed. and C. Benfield, Ed., "HTTP/2", RFC 9113, DOI 10.17487/RFC9113, June 2022, <https://www.rfc-editor.org/rfc/rfc9113>.

[HTTP/3] Bishop, M., Ed., "HTTP/3", RFC 9114, DOI 10.17487/RFC9114, June 2022, <https://www.rfc-editor.org/rfc/rfc9114>.

[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/rfc/rfc3552>.

[RFC6265] Barth, A., "HTTP State Management Mechanism", RFC 6265, DOI 10.17487/RFC6265, April 2011, <https://www.rfc-editor.org/rfc/rfc6265>.

[SNIP] Thomson, M., "Secure Negotiation of Incompatible Protocols in TLS", Work in Progress, Internet-Draft, draft-ietf-tls-snip-02, 30 June 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-tls-snip-02>.

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

Appendix A. Authoritative Information in RFC 7838

This design differs from RFC 7838, where alternative services advertisements were treated as authoritative information. Clients therefore might have been less concerned about attacks that compromise the integrity of alternative services when using RFC 7838.

Though integrity protection might appear to be valuable, it results in conflicts. For instance, information about the protocol is ostensibly authentic when provided in Alt-Svc fields or ALTSVC frames. However, protocol support is also authenticated when establishing a connection. This creates a potential conflict between two sources of the same information.

Conflicts also arise when alternative service information is retained as any retained state might disagree with what is currently deployed. This design avoids this contention by delegating the service resolution process almost entirely to the DNS.

This design provides clients with a prompt to discover a new service endpoint. On subsequent connections, remembered state only affects prioritization of active DNS records. Service endpoints are always authoritative for their own configuration. Invalid configurations therefore do not persist.

Contributors

RFC 7838 [ALT-SVC] was authored by Patrick McManus, Julian Reschke, and Mark Nottingham. This draft contains none of that work, but many of those same basic ideas.

Acknowledgments

This work is input to discussions with a design team on HTTP alternative services, formed after realizing that a simple revision to RFC 7838 would not fix known problems. Thanks are due to Ryan Hamilton, Tommy Pauly, and Matthew Stock for their contributions to these discussions. David Schinazi also provided valuable input.

Index

Section 1.2, Paragraph 2; Section 1.2, Paragraph 5;
Section 1.3, Paragraph 2; Section 1.3, Paragraph 3;
Section 1.3, Paragraph 4; Section 1.3, Paragraph 5;
Section 2, Paragraph 1; Section 2, Paragraph 1;
Section 2, Paragraph 2; Section 2, Paragraph 2;
Section 2, Paragraph 3; Section 2, Paragraph 4, Item 2;
Section 2, Paragraph 4, Item 2;
Section 2, Paragraph 4, Item 2;
Section 2, Paragraph 4, Item 3;
Section 2, Paragraph 4, Item 3;
Section 2, Paragraph 4, Item 3;
Section 2, Paragraph 4, Item 4;
Section 2, Paragraph 4, Item 4; Section 2, Paragraph 5;
Section 2, Paragraph 6; Section 2, Paragraph 6;
Section 2.1, Paragraph 2, Item 1;
Section 2.1, Paragraph 2, Item 2; Section 2.1, Paragraph 3;
Section 2.1, Paragraph 3; Section 2.1, Paragraph 3;
Section 2.1, Paragraph 4; Section 2.1, Paragraph 5;
Section 2.1, Paragraph 6; Section 2.1, Paragraph 7;
Section 2.1, Paragraph 8; Section 2.2.1, Paragraph 4;
Section 2.2.2, Paragraph 1; Section 2.2.2, Paragraph 4;
Section 2.3, Paragraph 1; Section 2.5, Paragraph 2;
Section 2.6, Paragraph 2; Section 2.6, Paragraph 3;
Section 2.7, Paragraph 1; Section 2.8, Paragraph 1;
Section 2.8, Paragraph 1; Section 2.8, Paragraph 2;
Section 2.8, Paragraph 2; Section 2.8, Paragraph 3;
Section 3, Paragraph 1; Section 3.2, Paragraph 6;
Section 4.2, Paragraph 1; Section 4.3, Paragraph 1;
Section 4.3, Paragraph 2

service endpoint

Section Abstract, Paragraph 1;
Section Abstract, Paragraph 1; Section 1, Paragraph 1;
Section 1, Paragraph 2; Section 1, Paragraph 2;
Section 1, Paragraph 3, Item 1; Section 1, Paragraph 4;
Section 1, Paragraph 4; Section 1, Paragraph 5;
Section 1.1, Paragraph 2; Section 1.1, Paragraph 3;
Section 1.3, Paragraph 3; Section 1.3, Paragraph 4;
Section 1.3, Paragraph 4; Section 1.3, Paragraph 5;
Section 2, Paragraph 2; Section 2, Paragraph 3;
Section 2, Paragraph 4, Item 5; Section 2.8, Paragraph 1;
Section 2.8, Paragraph 3; Section 2.8, Paragraph 3;
Section 4, Paragraph 1; Section 4, Paragraph 1;
Section 4.1, Paragraph 2; Section 4.1, Paragraph 2;
Section 4.1, Paragraph 3; Section 4.1, Paragraph 3;
Section 4.1, Paragraph 4; Section 4.1, Paragraph 5;
Section 4.2, Paragraph 2; Section 4.4, Paragraph 1;

Authors' Addresses

Martin Thomson
Mozilla

Email: mt@lowentropy.net

Mike Bishop
Akamai Technologies

Email: mbishop@evequefou.be

Lucas Pardue
Cloudflare

Email: lucaspardue.24.7@gmail.com

Tommy Jensen
Microsoft

Email: tojens@microsoft.com