
Stream: Internet Engineering Task Force (IETF)
RFC: [9532](#)
Category: Standards Track
Published: January 2024
ISSN: 2070-1721
Author: T. Pauly
Apple, Inc.

RFC 9532

HTTP Proxy-Status Parameter for Next-Hop Aliases

Abstract

This document defines the `next-hop-aliases` HTTP Proxy-Status Parameter. This parameter carries the list of aliases and canonical names an intermediary received during DNS resolution as part of establishing a connection to the next hop.

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at <https://www.rfc-editor.org/info/rfc9532>.

Copyright Notice

Copyright (c) 2024 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

1. Introduction	2
1.1. Requirements	3
2. next-hop-aliases Parameter	3
2.1. Encoding Special Characters	4
3. Implementation Considerations	5
4. Security Considerations	5
5. IANA Considerations	5
6. References	6
6.1. Normative References	6
6.2. Informative References	6
Author's Address	7

1. Introduction

The Proxy-Status HTTP response field [[PROXY-STATUS](#)] allows intermediaries to convey information about how they handled the request in HTTP responses sent to clients. It defines a set of parameters that provide information, such as the name of the next hop.

[[PROXY-STATUS](#)] defines a `next-hop` parameter, which can contain a hostname, IP address, or alias of the next hop. This parameter can contain only one such item, so it cannot be used to communicate a chain of aliases encountered during DNS resolution when connecting to the next hop.

Knowing the full chain of names that were used during DNS resolution via CNAME records [[DNS](#)] is particularly useful for clients of forward proxies, in which the client is requesting to connect to a specific target hostname using the `CONNECT` method [[HTTP](#)] or UDP proxying [[CONNECT-UDP](#)]. CNAME records can be used to "cloak" hosts that perform tracking or malicious activity behind more innocuous hostnames, and clients such as web browsers use the chain of DNS names to influence behavior like cookie usage policies [[COOKIES](#)] or the blocking of malicious hosts.

This document allows clients to receive the CNAME chain of DNS names for the next hop by including the list of names in a new `next-hop-aliases` Proxy-Status parameter.

1.1. Requirements

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. next-hop-aliases Parameter

The value of the `next-hop-aliases` parameter is a String [STRUCTURED-FIELDS] that contains one or more DNS names in a comma-separated list. The items in the list include all alias names and canonical names received in CNAME records [DNS] during the course of resolving the next hop's hostname using DNS and **MAY** include the original requested hostname itself. The names ought to appear in the order in which they were received in DNS, for the sake of consistency. If there are multiple CNAME records in the chain, the first name in the `next-hop-aliases` list would be the value in the CNAME record for the original hostname, and the final name in the `next-hop-aliases` list would be the name that ultimately resolved to one or more addresses.

The list of DNS names in `next-hop-aliases` parameter uses a comma (",") as a separator between names. Note that if a comma is included in a name itself, the comma must be encoded as described in [Section 2.1](#).

For example, consider a proxy "proxy.example.net" that receives the following records when performing DNS resolution for the next hop "host.example.com":

```
host.example.com.      CNAME  tracker.example.com.
tracker.example.com.  CNAME  service1.example.com.
service1.example.com. AAAA    2001:db8::1
```

The proxy could include the following proxy status in its response:

```
Proxy-Status: proxy.example.net; next-hop="2001:db8::1";
              next-hop-aliases="tracker.example.com,service1.example.com"
```

This indicates that "proxy.example.net", which used the IP address "2001:db8::1" as the next hop for this request, encountered the names "tracker.example.com" and "service1.example.com" in the DNS resolution chain. Note that while this example includes both the `next-hop` and `next-hop-aliases` parameters, `next-hop-aliases` can be included without including `next-hop`.

The proxy can also include the name of the next hop as the first item in the list. This is particularly useful for reverse proxies when clients would not otherwise know the name of the next hop, and the `next-hop` header is used to convey an IP address.

For example, consider a proxy "reverseproxy.example.net" that receives the following records when performing DNS resolution for the next hop "host.example.com":

```
host2.example.com.      CNAME  service2.example.com.  
service2.example.com.  AAAA   2001:db8::2
```

The proxy could include the following proxy status in its response:

```
Proxy-Status: reverseproxy.example.net; next-hop="2001:db8::2";  
             next-hop-aliases="host2.example.com, service2.example.com"
```

The `next-hop-aliases` parameter only applies when DNS was used to resolve the next hop's name, and it does not apply in all situations. Clients can use the information in this parameter to determine how to use the connection established through the proxy, but they need to gracefully handle situations in which this parameter is not present.

The proxy **MAY** send the empty string ("") as the value of `next-hop-aliases` parameter to indicate that no CNAME records were encountered when resolving the next hop's name.

2.1. Encoding Special Characters

DNS names commonly contain just alphanumeric characters and hyphens ("-"), although they are allowed to contain any character ([RFC1035], Section 3.1), including a comma. To prevent commas or other special characters in names leading to incorrect parsing, any characters that appear in names in this list that do not belong to the set of URI unreserved characters ([RFC3986], Section 2.3) **MUST** be percent-encoded as defined in [RFC3986], Section 2.1.

For example, consider the DNS name "comma.name.example.com". This name would be encoded within a `next-hop-aliases` parameter as follows:

```
Proxy-Status: proxy.example.net; next-hop="2001:db8::1";  
             next-hop-aliases="comma%2Cname.example.com, service1.example.com"
```

It is also possible for a DNS name to include a period character (".") within a label instead of as a label separator. In this case, the period character **MUST** first be escaped as "\.". Since the "\" character itself will be percent-encoded, the name "dot.label.example.com" would be encoded within a `next-hop-aliases` parameter as follows:

```
Proxy-Status: proxy.example.net; next-hop="2001:db8::1";  
             next-hop-aliases="dot%5C.label.example.com, service1.example.com"
```

Upon parsing this name, "dot%5C.label" **MUST** be treated as a single label.

Similarly, the "\" character in a label **MUST** be escaped as "\\" and then percent-encoded. Other uses of "\" **MUST NOT** appear in the label after percent-decoding. For example, if there is a DNS name "backslash.name.example.com", it would first be escaped as "backslash\\name.example.com" and then percent-encoded as follows:

```
Proxy-Status: proxy.example.net; next-hop="2001:db8::1";  
next-hop-aliases="backslash%5C%5Cname.example.com,s1.example.com"
```

3. Implementation Considerations

In order to include the `next-hop-aliases` parameter, a proxy needs to have access to the chain of alias names and canonical names received in CNAME records.

Implementations ought to note that the full chain of names might not be available in common DNS resolution APIs, such as `getaddrinfo` [POSIX]. `getaddrinfo` does have an option for `AI_CANONNAME` ([RFC3493], Section 6.1), but this will only return the last name in the chain (the canonical name), not the alias names.

An implementation **MAY** include incomplete information in the `next-hop-aliases` parameter to accommodate cases where it is unable to include the full chain, such as only including the canonical name if the implementation can only use `getaddrinfo` as described above.

4. Security Considerations

The `next-hop-aliases` parameter does not include any DNSSEC information or imply that DNSSEC was used. The information included in the parameter can only be trusted to be valid insofar as the client trusts the proxy to provide accurate information. This information is intended to be used as a hint and **SHOULD NOT** be used for making security decisions about the identity of a resource accessed through the proxy.

Inspecting CNAME chains can be used to detect cloaking of trackers or malicious hosts. However, the CNAME records could be omitted by a recursive or authoritative resolver that is trying to hide this form of cloaking. In particular, recursive or authoritative resolvers can omit these records for both clients directly performing DNS name resolution and proxies performing DNS name resolution on behalf of a client. A malicious proxy could also choose to not report these CNAME chains in order to hide the cloaking. In general, clients can trust information included (or not included) in the `next-hop-aliases` parameter to the degree that the proxy and any resolvers used by the proxy are trusted.

5. IANA Considerations

This document registers the `next-hop-aliases` parameter in the "HTTP Proxy-Status Parameters" registry <<https://www.iana.org/assignments/http-proxy-status>> as shown in Table 1.

Name	Description	Reference
next-hop-aliases	A string containing one or more DNS aliases or canonical names used to establish a proxied connection to the next hop.	RFC 9532

Table 1: HTTP Proxy-Status Parameters Registry

6. References

6.1. Normative References

- [CONNECT-UDP]** Schinazi, D., "Proxying UDP in HTTP", RFC 9298, DOI 10.17487/RFC9298, August 2022, <<https://www.rfc-editor.org/info/rfc9298>>.
- [DNS]** Mockapetris, P., "Domain names - concepts and facilities", STD 13, RFC 1034, DOI 10.17487/RFC1034, November 1987, <<https://www.rfc-editor.org/info/rfc1034>>.
- [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/info/rfc9110>>.
- [PROXY-STATUS]** Nottingham, M. and P. Sikora, "The Proxy-Status HTTP Response Header Field", RFC 9209, DOI 10.17487/RFC9209, June 2022, <<https://www.rfc-editor.org/info/rfc9209>>.
- [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>>.
- [RFC3986]** Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, DOI 10.17487/RFC3986, January 2005, <<https://www.rfc-editor.org/info/rfc3986>>.
- [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>>.
- [STRUCTURED-FIELDS]** Nottingham, M. and P. Kamp, "Structured Field Values for HTTP", RFC 8941, DOI 10.17487/RFC8941, February 2021, <<https://www.rfc-editor.org/info/rfc8941>>.

6.2. Informative References

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

- [POSIX]** IEEE, "IEEE Standard for Information Technology--Portable Operating System Interface (POSIX(TM)) Base Specifications, Issue 7", IEEE Std 1003.1-2017, DOI 10.1109/IEEESTD.2018.8277153, January 2018, <<https://ieeexplore.ieee.org/document/8277153>>.
- [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>>.
- [RFC3493]** Gilligan, R., Thomson, S., Bound, J., McCann, J., and W. Stevens, "Basic Socket Interface Extensions for IPv6", RFC 3493, DOI 10.17487/RFC3493, February 2003, <<https://www.rfc-editor.org/info/rfc3493>>.

Author's Address

Tommy Pauly

Apple, Inc.

Email: tpauly@apple.com