Workgroup: Network Working Group
Internet-Draft:
draft-schinazi-httpbis-transport-auth-01
Published: 8 January 2020
Intended Status: Experimental
Expires: 11 July 2020
Authors: D. Schinazi
Google LLC

HTTP Transport Authentication

Abstract

The most common existing authentication mechanisms for HTTP are sent with each HTTP request, and authenticate that request instead of the underlying HTTP connection, or transport. While these mechanisms work well for existing uses of HTTP, they are not suitable for emerging applications that multiplex non-HTTP traffic inside an HTTP connection. This document describes the HTTP Transport Authentication Framework, a method of authenticating not only an HTTP request, but also its underlying transport.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

The most common existing authentication mechanisms for HTTP are sent with each HTTP request, and authenticate that request instead of the underlying HTTP connection, or transport. While these mechanisms work well for existing uses of HTTP, they are not suitable for emerging applications that multiplex non-HTTP traffic inside an HTTP connection. This document describes the HTTP Transport Authentication Framework, a method of authenticating not only an HTTP request, but also its underlying transport.

Traditional HTTP semantics specify that HTTP is a stateless protocol where each request can be understood in isolation [RFC7230]. However, the emergence of QUIC [I-D.ietf-quic-transport] as a new transport protocol that can carry HTTP [I-D.ietf-quic-http] and the existence of QUIC extensions such as the DATAGRAM frame [I-D.pauly-quic-datagram] enable new uses of HTTP such as [I-D.vvv-webtransport-http3] and [I-D.schinazi-masque] where some traffic is exchanged that is disctinct from HTTP requests and responses. In order to authenticate this traffic, it is necessary to authenticate the underlying transport (e.g., QUIC or TLS [RFC8446]) instead of authenticate each request individually. This mechanism aims to supplement the HTTP Authentication Framework [RFC7235], not replace it.

Note that there is currently no mechanism for origin servers to request that user agents authenticate themselves using Transport Authentication, this is left as future work.

1.1. 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.

This document uses the Augmented BNF defined in [RFC5234] and updated by [RFC7405] along with the "#rule" extension defined in Section 7 of [RFC7230]. The rules below are defined in [RFC3061], [RFC5234], [RFC7230], and [RFC7235]:

```
OWS           = <OWS, see {{RFC7230}}, Section 3.2.3>
quoted-string = <quoted-string, see {{RFC7230}}, Section 3.2.6>
token         = <token, see {{RFC7230}}, Section 3.2.6>
token68       = <token, see {{RFC7235}}, Section 2.1>
oid           = <oid, see {{RFC3061}}, Section 2>
```

2. Computing the Authentication Proof

This document only defines Transport Authentication for uses of HTTP with TLS. This includes any use of HTTP over TLS as typically used for HTTP/2, or HTTP/3 where the transport protocol uses TLS as its authentication and key exchange mechanism [I-D.ietf-quic-tls].

The user agent leverages a TLS keying material exporter [RFC5705] to generate a nonce which can be signed using the user-id's key. The keying material exporter uses a label that starts with the characters "EXPORTER-HTTP-Transport-Authentication-" (see Section 4 for the labels and contexts used by each scheme). The TLS keying material exporter is used to generate a 32-byte key which is then used as a nonce.

3. Header Field Definition

The "Transport-Authentication" header allows a user agent to authenticate its transport connection with an origin server.

```
Transport-Authentication = transp-auth-scheme *( OWS ";" OWS parameter
transp-auth-scheme       = token
parameter                = token "=" ( token / quoted-string )
```

3.1. The u Directive

The OPTIONAL "u" (user-id) directive specifies the user-id that the user agent wishes to authenticate. It is encoded using Base64 (Section 4 of [RFC4648]).

```
u = token68
```

3.2. The p Directive

The OPTIONAL "p" (proof) directive specifies the proof that the user agent provides to attest to possessing the credential that matches its user-id. It is encoded using Base64 (Section 4 of [RFC4648]).

```
p = token68
```

3.3. The a Directive

The OPTIONAL "a" (algorithm) directive specifies the algorithm used to compute the proof transmitted in the "p" directive.

```
a = oid
```

4. Transport Authentication Schemes

The Transport Authentication Framework allows defining Transport Authentication Schemes, which specify how to authenticate user-ids. This documents defined the "Signature" and "HMAC" schemes.

4.1. Signature

The "Signature" Transport Authentication Scheme uses asymmetric cyptography. User agents possess a user-id and a public/private key pair, and origin servers maintain a mapping of authorized user-ids to their associated public keys. When using this scheme, the "u", "p", and "a" directives are REQUIRED. The TLS keying material export label for this scheme is "EXPORTER-HTTP-Transport-Authentication-Signature" and the associated context is empty. The nonce is then signed using the selected asymmetric signature algorithm and transmitted as the proof directive.

For example, the user-id "john.doe" authenticating using Ed25519 [RFC8410] could produce the following header (lines are folded to fit):

```
Transport-Authentication: Signature u="am9obi5kb2U=";a=1.3.101.112;
p="SW5zZXJ0IHNpZ25hdHVyZSBvZiBub25jZSBoZXJlIHdo
aWNoIHRha2VzIDUxMiBiaXRzIGZvciBFZDI1NTE5IQ=="
```

4.2. HMAC

The "HMAC" Transport Authentication Scheme uses symmetric cyptography. User agents possess a user-id and a secret key, and origin servers maintain a mapping of authorized user-ids to their associated secret key. When using this scheme, the "u", "p", and "a" directives are REQUIRED. The TLS keying material export label for this scheme is "EXPORTER-HTTP-Transport-Authentication-HMAC" and the associated context is empty. The nonce is then HMACed using the selected HMAC algorithm and transmitted as the proof directive.

For example, the user-id "john.doe" authenticating using HMAC-SHA-512 [RFC6234] could produce the following header (lines are folded to fit):

```
Transport-Authentication: HMAC u="am9obi5kb2U=";a=2.16.840.1.101.3.4.2.3
p="SW5zZXJ0IEhNQUMgb2Ygbm9uY2UgaGVyZSB3aGljaCB0YWtl
cyA1MTIgYml0cyBmb3IgU0hBLTUxMiEhISEhIQ=="
```

5. Proxy Considerations

Since Transport Authentication authenticates the underlying transport by leveraging TLS keying material exporters, it cannot be transparently forwarded by proxies that terminate TLS. However it can be sent over proxied connections when TLS is performed end-to-end (e.g., when using HTTP CONNECT proxies).

6. Security Considerations

Transport Authentication allows a user-agent to authenticate to an origin server while guaranteeing freshness and without the need for the server to transmit a nonce to the user agent. This allows the server to accept authenticated clients without revealing that it supports or expects authentication for some resources. It also allows authentication without the user agent leaking the presence of authentication to observers due to clear-text TLS Client Hello extensions.

7. IANA Considerations

7.1. Transport-Authentication Header Field

This document, if approved, requests IANA to register the "Transport-Authentication" header in the "Permanent Message Header Field Names" registry maintained at https://www.iana.org/assignments/message-headers/.

Header Field Name	Protocol	Status	Reference
Transport-Authentication	http	experimental	This document

7.2. Transport Authentication Schemes Registry

This document, if approved, requests IANA to create a new HTTP Transport Authentication Schemes Registry with the following entries:

Transport Authentication Scheme	Reference
Signature	This document
HMAC	This document

7.3. TLS Keying Material Exporter Labels

This document, if approved, requests IANA to register the following entries in the "TLS Exporter Labels" registry maintained at https://www.iana.org/assignments/tls-parameters/tls-parameters.xhtml#exporter-labels

Value
EXPORTER-HTTP-Transport-Authentication-Signature
EXPORTER-HTTP-Transport-Authentication-HMAC

Both of these entries are listed with the following qualifiers:

DTLS-OK	Recommended	Reference
N	Y	This document

8. References

8.1. Normative References

[RFC7230] Fielding, R., Ed. and J. Reschke, Ed., "Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing", RFC 7230, DOI 10.17487/RFC7230, June 2014, <https://www.rfc-editor.org/info/rfc7230>.

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

[RFC7235] Fielding, R., Ed. and J. Reschke, Ed., "Hypertext Transfer Protocol (HTTP/1.1): Authentication", RFC 7235, DOI 10.17487/RFC7235, June 2014, <https://www.rfc-editor.org/info/rfc7235>.

[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>.

[RFC5234] Crocker, D., Ed. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, DOI

10.17487/RFC5234, January 2008, <https://www.rfc-editor.org/info/rfc5234>.

[RFC7405] Kyzivat, P., "Case-Sensitive String Support in ABNF", RFC 7405, DOI 10.17487/RFC7405, December 2014, <https://www.rfc-editor.org/info/rfc7405>.

[RFC3061] Mealling, M., "A URN Namespace of Object Identifiers", RFC 3061, DOI 10.17487/RFC3061, February 2001, <https://www.rfc-editor.org/info/rfc3061>.

[RFC5705] Rescorla, E., "Keying Material Exporters for Transport Layer Security (TLS)", RFC 5705, DOI 10.17487/RFC5705, March 2010, <https://www.rfc-editor.org/info/rfc5705>.

[RFC4648] Josefsson, S., "The Base16, Base32, and Base64 Data Encodings", RFC 4648, DOI 10.17487/RFC4648, October 2006, <https://www.rfc-editor.org/info/rfc4648>.

8.2. Informative References

[I-D.ietf-quic-transport] Iyengar, J. and M. Thomson, "QUIC: A UDP-Based Multiplexed and Secure Transport", Work in Progress, Internet-Draft, draft-ietf-quic-transport-24, 3 November 2019, <http://www.ietf.org/internet-drafts/draft-ietf-quic-transport-24.txt>.

[I-D.ietf-quic-http] Bishop, M., "Hypertext Transfer Protocol Version 3 (HTTP/3)", Work in Progress, Internet-Draft, draft-ietf-quic-http-24, 4 November 2019, <http://www.ietf.org/internet-drafts/draft-ietf-quic-http-24.txt>.

[I-D.pauly-quic-datagram] Pauly, T., Kinnear, E., and D. Schinazi, "An Unreliable Datagram Extension to QUIC", Work in Progress, Internet-Draft, draft-pauly-quic-datagram-05, 4 November 2019, <http://www.ietf.org/internet-drafts/draft-pauly-quic-datagram-05.txt>.

[I-D.vvv-webtransport-http3] Vasiliev, V., "WebTransport over HTTP/3", Work in Progress, Internet-Draft, draft-vvv-webtransport-http3-01, 3 November 2019, <http://www.ietf.org/internet-drafts/draft-vvv-webtransport-http3-01.txt>.

[I-D.schinazi-masque] Schinazi, D., "The MASQUE Protocol", Work in Progress, Internet-Draft, draft-schinazi-masque-01, 8 July 2019, <http://www.ietf.org/internet-drafts/draft-schinazi-masque-01.txt>.

[I-D.ietf-quic-tls]
Thomson, M. and S. Turner, "Using TLS to Secure QUIC", Work in Progress, Internet-Draft, draft-ietf-quic-tls-24, 3 November 2019, <http://www.ietf.org/internet-drafts/draft-ietf-quic-tls-24.txt>.

[RFC8410] Josefsson, S. and J. Schaad, "Algorithm Identifiers for Ed25519, Ed448, X25519, and X448 for Use in the Internet X.509 Public Key Infrastructure", RFC 8410, DOI 10.17487/RFC8410, August 2018, <https://www.rfc-editor.org/info/rfc8410>.

[RFC6234] Eastlake 3rd, D. and T. Hansen, "US Secure Hash Algorithms (SHA and SHA-based HMAC and HKDF)", RFC 6234, DOI 10.17487/RFC6234, May 2011, <https://www.rfc-editor.org/info/rfc6234>.

[RFC7427] Kivinen, T. and J. Snyder, "Signature Authentication in the Internet Key Exchange Version 2 (IKEv2)", RFC 7427, DOI 10.17487/RFC7427, January 2015, <https://www.rfc-editor.org/info/rfc7427>.

Acknowledgments

The authors would like to thank many members of the IETF community, as this document is the fruit of many hallway conversations. Using the OID for the signature and HMAC algorithms was inspired by Signature Authentication in IKEv2 [RFC7427].

Author's Address

David Schinazi
Google LLC
1600 Amphitheatre Parkway
Mountain View, California 94043,
United States of America

Email: dschinazi.ietf@gmail.com