Internet Engineering Task Force
Internet-Draft
Intended status: Standards Track
Expires: August 29, 2018

B. Campbell
Ping Identity

February 25, 2018

HTTPS Token Binding with TLS Terminating Reverse Proxies
draft-ietf-tokbind-ttrp-03

Abstract

This document defines HTTP header fields that enable a TLS terminating reverse proxy to convey information to a backend server about the validated Token Binding Message received from a client, which enables that backend server to bind, or verify the binding of, cookies and other security tokens to the client's Token Binding key.

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 August 29, 2018.

Table of Contents

1. Introduction

Token Binding over HTTP [I-D.ietf-tokbind-https] provides a mechanism that enables HTTP servers to cryptographically bind cookies and other security tokens to a key held by the browser or other HTTP client, possession of which is proven on the TLS [RFC5246] connections over which the tokens are used. When Token Binding is negotiated in the TLS handshake [I-D.ietf-tokbind-negotiation] the client sends an encoded Token Binding Message [I-D.ietf-tokbind-protocol] as a header in each HTTP request, which proves possession of one or more private keys held by the client. The public portion of the keys are represented in the Token Binding IDs of the Token Binding Message and for each one there is a signature over some data, which includes the exported keying material [RFC5705] of the TLS connection. An HTTP server issuing cookies or other security tokens can associate them with the Token Binding ID, which ensures those tokens cannot be used successfully over a different TLS connection or by a different client than the one to which they were issued.

A fairly common deployment architecture for HTTPS applications is to have the backend HTTP application servers sit behind a reverse proxy that terminates TLS. The proxy is accessible to the internet and dispatches client requests to the appropriate backend server within a private or protected network. The backend servers are not directly accessible outside the private network and are only reachable through

the reverse proxy. The details of such deployments are typically opaque to clients who make requests to the proxy server and see responses as though they originated from the proxy server itself. TLS connections for HTTPS are established between each client and the reverse proxy server.

Token Binding facilitates a binding of security tokens to a key held by the client by way of the TLS connection between that client and the server. In a deployment where TLS is terminated by a reverse proxy, however, the TLS connection is between the client and the proxy while the backend server is likely the system that will issue and validate cookies or other security tokens. Additional steps are therefore needed to enable the use of Token Binding in such deployment architectures. In the absence of a standardized approach, different implementations will address it differently, which will make interoperability between such implementations difficult or impossible without complex configurations or custom integrations.

This document standardizes HTTP header field names that a TLS terminating reverse proxy (TTRP) adds to requests that it sends to the backend servers. The headers contain information from the validated Token Binding Message sent by the client to the proxy, thus enabling the backend server to bind, or verify the binding of, cookies and other security tokens to the client's Token Binding key. The usage of the headers, both the reverse proxy adding it and the application server using them to bind cookies or other tokens, are to be configuration options of the respective systems as they will not always be applicable.

1.1. Requirements Notation and Conventions

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. HTTP Header Fields and Processing Rules

2.1. Encoding

The field-values of the HTTP headers defined herein utilize the following encoded forms.

2.1.1. Token Binding ID

A Token Binding ID is represented as an "EncodedTokenBindingID", which is thea base64url encoding of the TokenBindingID byte sequence (see section 3 of [I-D.ietf-tokbind-protocol]) using the URL and filename safe alphabet described in Section 5 of [RFC4648], with all trailing pad characters '=' omitted and without the inclusion of any line breaks, whitespace, or other additional characters. ABNF [RFC5234] syntax for "EncodedTokenBindingID" is shown in Figure 1 below.

```
EncodedTokenBindingID = *( DIGIT / ALPHA /  "-" / "_" )

DIGIT = <Defined in Section B.1 of [RFC5234]>
ALPHA = <Defined in Section B.1 of [RFC5234]>
```

Figure 1: Encoded Token Binding ID ABNF

2.1.2. Token Binding Type

A Token Binding type can be represented as an "EncodedTokenBindingType", which is the base16 encoding (Section 8 of [RFC4648]) of the single TokenBindingType byte. ABNF [RFC5234] syntax for "EncodedTokenBindingType" is shown in Figure 2 below.

```
EncodedTokenBindingType = 1*2( DIGIT /
                               "A" / "B" / "C" / "D" / "E" / "F" /
                               "a" / "b" / "c" / "d" / "e" / "f" )
```

Figure 2: Encoded Token Binding Type ABNF

2.2. Token Binding ID HTTP Header Fields

The Token Binding Protocol [I-D.ietf-tokbind-protocol] recommends that implementations make Token Binding IDs available to the application as opaque byte sequences, enabling those applications to use the Token Binding IDs when generating and verifying bound tokens. In the context of a TLS terminating reverse proxy (TTRP) deployment, the TTRP makes the Token Binding ID(s) available to the backend application with the following header fields.

Sec-Provided-Token-Binding-ID
: The Token Binding ID of the provided Token Binding represented as an "EncodedTokenBindingID".

Sec-Referred-Token-Binding-ID
: The Token Binding ID of the referred Token Binding represented as an "EncodedTokenBindingID".

Sec-Other-Token-Binding-ID
: Additional Token Bindings, other than provided and referred, that are sent by the client and validated by the TTRP are represented as a comma-separated list of the concatenation of the "EncodedTokenBindingType", a period (".") character, and the "EncodedTokenBindingID" of each.

Both "Sec-Provided-Token-Binding-ID" and "Sec-Referred-Token-Binding-ID" are single HTTP header field-valued as defined in Section 3.2 of [RFC7230], which MUST NOT have a list of values or occur multiple times in a request.

All header fields defined herein are only for use in HTTP requests and MUST NOT to be used in HTTP responses.

2.3. Processing Rules

This section defines the applicable processing rules for a TLS terminating reverse proxy (TTRP) and backend server(s) to provide server side support of Token Binding over HTTP [I-D.ietf-tokbind-https] using the HTTP headers described in Section 2.2. Use of the technique is to be a configuration or deployment option and the processing rules described herein are for servers operating with that option enabled.

A TTRP negotiates the use of Token Binding with the client per [I-D.ietf-tokbind-negotiation] and validates the Token Binding Message as defined in The Token Binding Protocol [I-D.ietf-tokbind-protocol] and Token Binding over HTTP [I-D.ietf-tokbind-https] for each HTTP request on the underlying TLS connection. Requests with a valid Token Binding Message (and meeting any other authorization or policy requirements of the TTRP) are dispatched to the backend server with the following modifications.

1. The "Sec-Token-Binding" header in the original incoming request MUST be removed from the request that is dispatched to the backend server.

2. The Token Binding ID of the provided Token Binding of the Token Binding Message MUST be placed in the "Sec-Provided-Token-Binding-ID" header field of the dispatched request using the format defined in Section 2.2.

3. If the Token Binding Message contains a referred Token Binding, the referred Token Binding ID MUST be placed in the "Sec-Referred-Token-Binding-ID" header field of the dispatched request using the format defined in Section 2.2. Otherwise, the "Sec-

Referred-Token-Binding-ID" header field MUST NOT be present in the dispatched request.

4. If the Token Binding Message contains any additional validated Token Bindings, they are placed in the "Sec-Other-Token-Binding-ID" header field using the format defined in Section 2.2. If the Token Binding Message contains no additional valid Token Bindings, the "Sec-Referred-Token-Binding-ID" header field MUST NOT be present in the dispatched request.

5. Any occurrence of the "Sec-Provided-Token-Binding-ID", "Sec-Referred-Token-Binding-ID", and "Sec-Other-Token-Binding-ID" headers in the original incoming request MUST be removed or overwritten before forwarding the request.

Requests made over a connection where the use of Token Binding was not negotiated MUST be sanitized by removing any occurrences of the "Sec-Provided-Token-Binding-ID", "Sec-Referred-Token-Binding-ID", and "Sec-Other-Token-Binding-ID" header fields prior to dispatching the request to the backend server.

Forward proxies and other intermediaries MUST NOT add the "Sec-Provided-Token-Binding-ID" "Sec-Referred-Token-Binding-ID", or "Sec-Other-Token-Binding-ID" header to requests.

2.4. Examples

Extra line breaks and whitespace have been added to the following examples for display and formatting purposes only.

2.4.1. Provided Token Binding ID

The following "Sec-Token-Binding" header is from an HTTP request made over a TLS connection between the client and the TTRP where the use of Token Binding has been negotiated (The base64url-encoded representation of the exported keying material, which can be used to validate the Token Binding Message, for that connection is "AYVUayPTP9RmELNpGjFl6Ykm2CUx7pUMxe35yb11dgU"). The encoded Token Binding Message has the provided Token Binding the client uses with the server.

```
Sec-Token-Binding: AIkAAgBBQKzyIrmcY_YCtHVoSHBut69vrGfFdy1_YKTZfFJv
 6BjrZsKD9b9FRzSBxDs1twTqnAS71M1RBumuihhI9xqxXKkAQEtxe4jeUJU0WezxlQ
 XWVSBFeHxFMdXRBIH_LKOSAuSMOJ0XEw1Q8DE248qkOiRKzw3KdSNYukYEPmO21bQi
 3YYAAA
```

Figure 3: Header in HTTP Request to TTRP

After validating the Token Binding Message, the TTRP removes the "Sec-Token-Binding" header and adds the following "Sec-Provided-Token-Binding-ID" header with the provided Token Binding ID to the request that is dispatched to the backend server.

```
Sec-Provided-Token-Binding-ID: AgBBQKzyIrmcY_YCtHVoSHBut69vrGfFdy1_
  YKTZfFJv6BjrZsKD9b9FRzSBxDs1twTqnAS71M1RBumuihhI9xqxXKk
```

Figure 4: Header in HTTP Request to Backend Server

2.4.2. Provided and Referred Token Binding IDs

The following "Sec-Token-Binding" header is from an HTTP request made over a TLS connection between the client and the TTRP where the use of Token Binding has been negotiated (The base64url-encoded representation of the exported keying material, which can be used to validate the Token Binding Message, for that connection is "wEWWCP1KPxfq-QL4NxYII_P4ti_9YYqrTpGs28BZEqE"). The encoded Token Binding Message has the provided Token Binding the client uses with the server as well as the referred Token Binding that it uses with a different server.

```
Sec-Token-Binding: ARIAAgBBQCfsI1D1sTq5mvT_2H_dihNIvuHJCHGjHPJchPav
  NbGrOo26-2JgT_IsbvZd4daDFbirYBIwJ-TK1rh8FzrC-psAQMyYIqXj7djGPev1dk
  jV9XxLYGCyqOrBVEtBHrMUCeo22ymLg3OiFcl_fmOPxJbjxI6lKcF0lyfy-dSQmPIe
  zQ0AAAECAEFArPIiuZxj9gK0dWhIcG63r2-sZ8V3LX9gpNl8Um_oGOtmwoP1v0VHNI
  HEOzW3BOqcBLvUzVEG6a6KGEj3GrFcqQBAHQm0pzgUTXKLRamuKE1pmmP9I3UBVpoe
  1DBCe9H2l1VPpsImakUa6crAqZ-0CGBmji7bYzQogpKcyxTTFk5zdwAA
```

Figure 5: Header in HTTP Request to TTRP

After validating the Token Binding Message, the TTRP removes the "Sec-Token-Binding" header and adds the following "Sec-Provided-Token-Binding-ID" and "Sec-Referred-Token-Binding-ID" headers, with the provided and referred Token Binding IDs respectively, to the request that is dispatched to the backend server.

```
Sec-Provided-Token-Binding-ID: AgBBQCfsI1D1sTq5mvT_2H_dihNIvuHJCHGj
  HPJchPavNbGrOo26-2JgT_IsbvZd4daDFbirYBIwJ-TK1rh8FzrC-ps
Sec-Referred-Token-Binding-ID: AgBBQKzyIrmcY_YCtHVoSHBut69vrGfFdy1_
  YKTZfFJv6BjrZsKD9b9FRzSBxDs1twTqnAS71M1RBumuihhI9xqxXKk
```

Figure 6: Headers in HTTP Request to Backend Server

3. Security Considerations

The headers described herein enable a reverse proxy and backend server to function together as though they are single logical server side deployment of HTTPS Token Binding. Use of the headers outside that intended use case, however, may undermine the protections afforded by Token Binding. Therefore steps MUST be taken to prevent unintended use, both in sending the headers and in relying on their value.

Producing and consuming the headers SHOULD be a configurable option, respectively, in a reverse proxy and backend server (or individual application in that server). The default configuration for both should be to not use the headers thus requiring an "opt-in" to the functionality.

Backend servers MUST only accept the headers from trusted reverse proxies. And reverse proxies MUST sanitize the incoming request before forwarding it on by removing or overwriting any existing instances of the headers. Otherwise arbitrary clients can control the header values as seen and used by the backend server.

The communication between a reverse proxy and backend server needs to be secured against eavesdropping and modification by unintended parties.

The configuration options and request sanitization are necessarily functionally of the respective servers. The other requirements can be met in a number of ways, which will vary based on specific deployments. The communication between a reverse proxy and backend server, for example, might be over a mutually authenticated TLS with the insertion and consumption headers occurring only on that connection. Alternatively the network topology might dictate a private network such that the backend application is only able to accept requests from the reverse proxy and the proxy can only make requests to that server. Other deployments that meet the requirements set forth herein are also possible.

Employing the "Sec-" header field prefix for the headers defined herein denotes them as forbidden header names (see [fetch-spec]), which means they cannot be set or modified programmatically by script running in-browser.

4. IANA Considerations

4.1. HTTP Message Header Field Names Registration

This document specifies the following new HTTP header fields, registration of which is requested in the "Permanent Message Header Field Names" registry defined in [RFC3864].

- Header Field Name: "Sec-Provided-Token-Binding-ID"
- Applicable protocol: HTTP
- Status: standard
- Author/change Controller: IETF
- Specification Document(s): [[this specification]]

- Header Field Name: "Sec-Referred-Token-Binding-ID"
- Applicable protocol: HTTP
- Status: standard
- Author/change Controller: IETF
- Specification Document(s): [[this specification]]

- Header Field Name: "Sec-Other-Token-Binding-ID"
- Applicable protocol: HTTP
- Status: standard
- Author/change Controller: IETF
- Specification Document(s): [[this specification]]

5. References

5.1. Normative References

[I-D.ietf-tokbind-https]
Popov, A., Nystrom, M., Balfanz, D., Langley, A., Harper, N., and J. Hodges, "Token Binding over HTTP", draft-ietf-tokbind-https-12 (work in progress), January 2018.

[I-D.ietf-tokbind-negotiation]
Popov, A., Nystrom, M., Balfanz, D., and A. Langley, "Transport Layer Security (TLS) Extension for Token Binding Protocol Negotiation", draft-ietf-tokbind-negotiation-10 (work in progress), October 2017.

[I-D.ietf-tokbind-protocol]
Popov, A., Nystrom, M., Balfanz, D., Langley, A., and J. Hodges, "The Token Binding Protocol Version 1.0", draft-ietf-tokbind-protocol-16 (work in progress), October 2017.

[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>.

[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>.

[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>.

[RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, DOI 10.17487/RFC5246, August 2008, <https://www.rfc-editor.org/info/rfc5246>.

[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>.

[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>.

[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>.

5.2. Informative References

[fetch-spec] WhatWG, "Fetch", Living Standard , <https://fetch.spec.whatwg.org/>.

[RFC3864] Klyne, G., Nottingham, M., and J. Mogul, "Registration Procedures for Message Header Fields", BCP 90, RFC 3864, DOI 10.17487/RFC3864, September 2004, <https://www.rfc-editor.org/info/rfc3864>.

Appendix A. Acknowledgements

The author would like to thank the following people for their various contributions to the specification: Vinod Anupam, Dirk Balfanz, John Bradley, William Denniss, Nick Harper, Jeff Hodges, Subodh Iyengar, Leif Johansson, Michael B. Jones, Yoav Nir, Andrei Popov, Eric Rescorla, Piotr Sikora, Martin Thomson, Hans Zandbelt and others (please let me know, if you've contributed and I've forgotten you).

Appendix B. Document History

[[to be removed by the RFC Editor before publication as an RFC]]

draft-ietf-tokbind-ttrp-03

- Add a header to allow for additional token binding types other than provided and referred to be conveyed.
- Reword the Abstract somewhat for (hopefully) improved readability.
- Minor editorial and formatting updates.

draft-ietf-tokbind-ttrp-02

- Add to the Acknowledgements.
- Update references for Token Binding negotiation, protocol, and https.
- Use the boilerplate from RFC 8174.
- Reformat the "HTTP Header Fields and Processing Rules" section to make the header names more prominent and move the encoding definitions earlier.

draft-ietf-tokbind-ttrp-01

- Prefix the header names with "Sec-" so that they are denoted as forbidden header names by Fetch https://fetch.spec.whatwg.org/
- Removed potentially confusing sentence from Security Considerations per https://mailarchive.ietf.org/arch/msg/unbearable/O0IpppyyEqMrQjEkyEi8p8CeBGA
- Editorial fixes.

draft-ietf-tokbind-ttrp-00

- Initial WG draft from draft-campbell-tokbind-ttrp.

draft-campbell-tokbind-ttrp-01

- Minor editorial fixes.
- Add to the Acknowledgements.

draft-campbell-tokbind-ttrp-00

- Initial draft based on 'consensus to work on the problem' from the Seoul meeting [1][2] and reflecting the consensus approach from discussions at the Chicago meeting [3].

  [1] https://www.ietf.org/proceedings/97/minutes/minutes-97-tokbind-01.txt (minutes from Seoul)
  [2] https://www.ietf.org/proceedings/97/slides/slides-97-tokbind-reverse-proxies-00.pdf (slides from Seoul)
  [3] https://mailarchive.ietf.org/arch/msg/unbearable/_ZHI8y2Vs5WMP8VMRr7zroo_sNU (summary of discussion)

Author's Address

Brian Campbell
Ping Identity

Email: brian.d.campbell@gmail.com