TLS
Internet-Draft
Intended status: Standards Track
Expires: September 26, 2019

Y. Nir
DellEMC

March 25, 2019

# A Flags Extension for TLS 1.3
# draft-nir-tls-tlsflags-00

Abstract

A number of extensions are proposed in the TLS working group that carry no interesting information except the 1-bit indication that a certain optional feature is supported. Such extensions take 4 octets each. This document defines a flags extension that can provide such indications with only 1 octet each.

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 September 26, 2019.

Copyright Notice

Copyright (c) 2019 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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction . . . . . 2
1.1. Requirements and Other Notation . . . . . 2
2. The tls_flags Extension . . . . . 3
3. IANA Considerations . . . . . 3
4. Security Considerations . . . . . 4
5. Acknowledgements . . . . . 4
6. References . . . . . 4
6.1. Normative References . . . . . 4
6.2. Informative References . . . . . 5
Author's Address . . . . . 5

## 1. Introduction

Since the publication of TLS 1.3 ([RFC8446]) there have been several proposal for extensions to this protocol, where the presence of the content-free extension in both the ClientHello and either the ServerHello or EncryptedExtensions indicates nothing except either support for the optional feature or an intent to use the optional feature. Examples:

- An extension that allows the server to tell the client that cross-SNI resumption is allowed: [I-D.sy-tls-resumption-group].

- An extension that is used to negotiate support for authentication using both certificates and external PSKs: [I-D.ietf-tls-tls13-cert-with-extern-psk].

This document proposes a single extension called tls_flags that can enumerate such flag extensions and allowing both client and server to indicate support for optional features in a concise way.

### 1.1. Requirements and Other Notation

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD 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 term "flag extension" is used to denote an extension where the extension_data field is zero-length in all cases and the presence of the extension denotes either support for some feature or the intent to use that feature.

The term "flag-type feature" denotes an options TLS 1.3 feature the support for which is negotiated using a flag extension, whether that

flag extension is its own extension or a value in the extension defined in this document.

## 2. The tls_flags Extension

This document defines the following extension code point:

```
enum {
   ...
   tls_flags(TBD),
   (65535)
} ExtensionType;
```

This document also defines the data for this extension as a list of enums. Note that this document does not define any values to FlagExtensionType. That is left to the actual protocol document such as the ones in the examples from the previous section.

```
enum {
   (255)
} FlagExtensionType;

struct {
   FlagExtensionType flags<1..2^8-1>
} FlagExtensions;
```

A client that supports this extension and also supports at least one of the flag-type features that use this extension SHALL send this extension with the FlagExtensions data containing all those extensions that it supports.

A server that supports this extension and also supports at least one of the flag-type features that use this extension and that were declared by the ClientHello extension SHALL send this extension with the intersection of the flags it supports with the flags declared by the client. The server may need to send two tls_flags extensions, one in the ServerHello and the other in the EncryptedExtensions message. It is up to the document for the specific feature to determine whether support should be acknowledged in the ServerHello or the EncryptedExtensions message.

## 3. IANA Considerations

IANA is requested to assign a new value from the TLS ExtensionType Values registry:

o The Extension Name should be tls_flags

- The TLS 1.3 value should be CH,SH,EE
- The Recommended value should be Y
- The Reference should be this document

IANA is also requested to create a new registry under the TLS namespace with name "TLS Flags" and the following fields:

- Value, which is a value between 0 and 255. The values 240-255 should be Reserved for Private Use, while the rest should be available for assignment.
- Flag Name, which is a string
- Message, which like the "TLS 1.3" field in the ExtensionType registry contains the abbreviations of the messages that may contain the flag: CH, SH, EE, etc.
- Recommended, which is a Y/N value determined in the document defining the optional feature
- Reference, which is a link to the document defining this flag.

The policy for this shall be "Specification Required" as described in [RFC8126].

## 4. Security Considerations

The extension described in this document provides a more concise way to express data that could otherwise be expressed in individual extensions. It does not send in the clear any information that would otherwise be sent encrypted, nor vice versa. For this reason this extension is neutral as far as security is concerned.

## 5. Acknowledgements

The idea for writing this was expressed at the mic during the TLS session at IETF 104 by EKR.

## 6. References

### 6.1. Normative References

[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>.

[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>.

### 6.2. Informative References

[I-D.ietf-tls-tls13-cert-with-extern-psk]
Housley, R., "TLS 1.3 Extension for Certificate-based Authentication with an External Pre-Shared Key", draft-ietf-tls-tls13-cert-with-extern-psk-00 (work in progress), February 2019.

[I-D.sy-tls-resumption-group]
Sy, E., "TLS Resumption across Server Name Indications for TLS 1.3", draft-sy-tls-resumption-group-00 (work in progress), March 2019.

[RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <https://www.rfc-editor.org/info/rfc8126>.

## Author's Address

Yoav Nir
DellEMC
9 Andrei Sakharov St
Haifa 3190500
Israel

Email: ynir.ietf@gmail.com