Transport Layer Security
Internet-Draft
Intended status: Standards Track
Expires: 3 September 2026

Y. Rosomakho
Zscaler
J. Hoyland
Cloudflare

2 March 2026

Workload Identifier Origin Hint for TLS ClientHello
draft-rosomakho-tls-wimse-cert-hint-02

Abstract

This document defines a TLS extension that allows clients to indicate one or more workload identifier origins in the ClientHello message. Each origin consists of a URI scheme and trust domain component, representing the administrative domain and identifier namespace in which the client operates. These identifier origins serve as hints to enable the server to determine whether client authentication is required and which policies or trust anchors should apply. This mechanism improves efficiency in mutual TLS deployments while minimising the exposure of sensitive identifier information. To protect confidentiality, this extension can be used in conjunction with Encrypted Client Hello (ECH).

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://yaroslavros.github.io/tls-wimse-cert-hint/draft-rosomakho-tls-wimse-cert-hint.html. Status information for this document may be found at https://datatracker.ietf.org/doc/draft-rosomakho-tls-wimse-cert-hint/.

Discussion of this document takes place on the Transport Layer Security mailing list (mailto:tls@ietf.org), which is archived at https://mailarchive.ietf.org/arch/browse/tls/. Subscribe at https://www.ietf.org/mailman/listinfo/tls/.

Source for this draft and an issue tracker can be found at https://github.com/yaroslavros/tls-wimse-cert-hint.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

This Internet-Draft will expire on 3 September 2026.

Copyright Notice

Table of Contents

1. Introduction

Mutual TLS (mTLS) is commonly used to authenticate both endpoints of a [TLS] connection, especially in service-to-service communication within distributed systems. In many deployments, client authentication is conditional: only certain clients are required to present a certificate, and the decision is based on the nature of the client.

This document defines a TLS extension that allows clients to indicate one or more workload identifier origins in the ClientHello message (Section 4.1.2 of [TLS]). As defined in Section 4.5 of [WIMSE-IDENTIFIER], workload identifier origin is a subset of workload identifier and consists of a URI scheme and a trust domain (e.g., spiffe://example.org or wimse://botfarm.example.com). It indicates a namespace under which the client may present an authenticated identifier. Workload identifier origins act as hints that inform the server of the client intended identifier before the TLS handshake is completed. Based on this information, the server can determine whether client certificate authentication is desirable and, if so, what policy or certificate validation rules should apply.

This approach enables more flexible and efficient authentication strategies in environments where different clients may be subject to different requirements. For example:

* A server may enforce mTLS only for clients of specific workload identifier origin and allow others to connect without client certificate authentication on TLS layer.

* A server may use the provided workload identifier origins to generate an appropriate list of Certificate Authorities extension (Section 4.2.4 of [TLS]) in CertificateRequest message (Section 4.3.2 of [TLS]).

* The server may reject the connection early if none of the advertised workload identifier origins are authorized.

By only sending scheme and trust domain (omitting the path), this extension limits exposure of cleartext information. Where further confidentiality is desired, clients are encouraged to include this extension only in ClientHelloInner of Encrypted Client Hello ([ECH]) to ensure confidentiality of the workload identifier origins.

2. 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.

3. TLS Extension Format

This document defines a new TLS extension named workload_identifier_origin_hint, which is carried in the ClientHello message. The extension provides the server with one or more workload identifier origins that the client associates with itself. This allows the server to evaluate authentication requirements prior to sending a CertificateRequest message.

The workload_identifier_origin_hint extension is structured as follows:

```
opaque WorkloadIdentifierOrigin<1..2^16-1>;

struct {
    WorkloadIdentifierOrigin identifierorigins<3..2^16-1>;
} WorkloadIdentifierOriginHintExtension;
```

identifierOrigins: A list of UTF-8 encoded absolute URI strings as defined in [URI] containing only the scheme and trust domain components of Workload Identifiers as defined in Section 4.5 of [WIMSE-IDENTIFIER]. URI strings MUST NOT contain a path component.

Clients MAY include multiple identity origins if they operate within more than one trust domain or namespace.

The extension MUST appear only in the ClientHello. Servers MUST abort TLS handshake with an illegal_parameter alert if this extension appears in any other handshake message. Similarly, clients MUST abort TLS handshake if this extension appears in any message from the server.

3.1. Server Processing Rules

Upon receiving the extension, the server:

* MAY use the identifier origins to determine whether to send a CertificateRequest message.

* MAY use the identifier origins to construct Certificate Authorities extension in the CertificateRequest message.

* MAY use the identifier origins to select a trust anchor or policy.

* MAY reject the handshake early with handshake_failure alert if none of the identifier origins are acceptable.

* MUST NOT treat inclusion of the extension as proof of identity. The identifier origins are advisory and unauthenticated until verified during client authentication.

If the extension is absent, the server proceeds with the default client authentication behavior.

4. Security Considerations

This extension is intended to improve the flexibility of client authentication policies in TLS. However, because it introduces unauthenticated identity hints early in the handshake, several security considerations apply.

4.1. Confidentiality of Workload Identifier Origins

Workload identifier origins may contain sensitive information, such as deployment structure or tenant-specific data. Since this extension is sent in the clear as part of the ClientHello, exposure of these identifier origins may allow passive observers to infer client roles, access patterns, or security posture.

To mitigate this risk, clients SHOULD include this extension only in ClinetHelloInner if [ECH] is available. ECH encrypts the ClinetHelloInner and its extensions under the server’s public key, preventing visibility of the identifier origins to on-path observers.

If ECH is not in use, clients SHOULD avoid including sensitive or detailed identifier origins in this extension unless required by policy.

4.2. Unauthenticated Hints

The workload identifier origins conveyed in this extension are not authenticated. They are advisory in nature and MUST NOT be treated by the server as a proof of identity. Servers MUST perform full cryptographic verification of the client certificate before relying on any identity claim.

Servers MAY enforce policies based on the presence or absence of expected identifier origins in the ClientHello. However, this enforcement must be restricted to access control decisions prior to authentication, such astriggering client authentication or rejecting the handshake.

4.3. Identifier Origins Enumeration

If ECH is not deployed, an attacker with network visibility may collect workload identifier origins by observing repeated TLS handshakes. This could aid in reconnaissance or allow inference of infrastructure details. To reduce this risk, clients may:

* Use generic or opaque identifier origins when full disclosure is not required.

* Limit use of the extension to trusted networks or peers.

* Use ECH to encrypt the extension contents.

4.4. Server Response Behaviour

Servers receiving unknown or malformed identifier origins SHOULD ignore them and proceed with the default authentication policy. Servers SHOULD NOT terminate connections solely due to unrecognised identifier origins unless explicitly configured to do so.

5. IANA Considerations

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

* The Extension Name should be workload_identifier_origin_hint

* The TLS 1.3 value should be CH

* The DTLS-Only value should be N

* The Recommended value should be Y

* The Reference should be this document

6. Normative References

[ECH] Rescorla, E., Oku, K., Sullivan, N., and C. A. Wood, "TLS Encrypted Client Hello", Work in Progress, Internet-Draft, draft-ietf-tls-esni-25, 14 June 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-tls-esni-25>.

[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>.

[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>.

[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>.

[URI] 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/rfc/rfc3986>.

[WIMSE-IDENTIFIER] Rosomakho, Y. and J. A. Salowey, "Workload Identifier", Work in Progress, Internet-Draft, draft-ietf-wimse-identifier-02, 2 March 2026, <https://datatracker.ietf.org/doc/html/draft-ietf-wimse-identifier-02>.

Acknowledgments

TODO acknowledge.

Authors' Addresses

Yaroslav Rosomakho
Zscaler
Email: yrosomakho@zscaler.com

Jonathan Hoyland
Cloudflare
Email: jonathan.hoyland@gmail.com