Workload Identity in Multi System Environments
Internet-Draft
Intended status: Standards Track
Expires: 6 November 2026

j Salowey
CyberArk
Y. Rosomakho
Zscaler

5 May 2026

Workload Authentication Using Mutual TLS
draft-ietf-wimse-mutual-tls-01

Abstract

The WIMSE architecture defines authentication and authorization for software workloads in a variety of runtime environments, from the most basic ones to complex multi-service, multi-cloud, multi-tenant deployments. This document profiles a workload authentication based on X.509 workload identity certificates using mutual TLS (mTLS).

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://ietf-wg-wimse.github.io/draft-ietf-wimse-s2s-protocol/draft-ietf-wimsemutual-tls.html. Status information for this document may be found at https://datatracker.ietf.org/doc/draft-ietf-wimse-mutual-tls/.

Discussion of this document takes place on the Workload Identity in Multi System Environments Working Group mailing list (mailto:wimse@ietf.org), which is archived at https://mailarchive.ietf.org/arch/browse/wimse/. Subscribe at https://www.ietf.org/mailman/listinfo/wimse/.

Source for this draft and an issue tracker can be found at https://github.com/ietf-wg-wimse/draft-ietf-wimse-s2s-protocol.

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/.

Copyright Notice

Table of Contents

1. Introduction

This document defines authentication and authorization in the context of interaction between two workloads. This is the core component of the WIMSE architecture [I-D.ietf-wimse-arch]. This document focuses on using X.509 workload identity certificates [I-D.ietf-wimse-workload-creds] to authenticate the communication between workloads using TLS.

The use of TLS for authentication is widely deployed, however it may not be applicable to all environments. For example, some deployments may lack the PKI infrastructure necessary to manage certificates or inter-service communication consists of multiple separate TLS hops. For these cases, other options based on Workload Identity Tokens (WIT) [I-D.ietf-wimse-workload-creds] may be more appropriate since they are not based on X.509 certificates and are communicated at the application layer rather than the transport layer.

1.1. Deployment Architecture and Message Flow

Refer to Sec. 1.2 of [I-D.ietf-wimse-workload-creds] for the deployment architecture which is common to all three protection options, including the one described here.

1.2. Workload Identity Certificates

Workload identity certificates are X.509 certificates that carry workload identifiers as described in section 4.1 of [I-D.ietf-wimse-workload-creds]

2. Conventions and Definitions

All terminology in this document follows [I-D.ietf-wimse-arch].

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. Using Mutual TLS for Workload-to-Workload Authentication

As noted in the introduction, for many deployments, transport-level protection of application traffic using TLS is ideal.

3.1. The Workload Identity Certificate

Workload identity certificates are X.509 certificates that carry workload identifiers as described in section 4.1 of [I-D.ietf-wimse-workload-creds]

3.2. Workload Identity Certificate Validation

Workload Identity Certificates may be used to authenticate both the server and client side of the connections. When validating a Workload Identity Certificate, the relying party MUST use the trust anchors configured for the trust domain in the workload identity to validate the peer's certificate. Other PKIX [RFC5280] path validation rules apply. Workloads acting as TLS clients and servers MUST validate that the trust domain portion of the Workload Identity Certificate matches the expected trust domain for the other side of the connection.

Servers wishing to use the Workload Identity Certificate for authorizing the client MUST require client certificate authentication in the TLS handshake. Other methods of post handshake authentication are not specified by this document.

Workload Identity Certificates used by TLS servers SHOULD have the id-kp-serverAuth extended key usage [RFC5280] field set and Workload Identity Certificates used by TLS clients SHOULD have the id-kp-clientAuth extended key usage field set. A certificate that is used for both client and server connections may have both fields set. This specification does not make any other requirements beyond [RFC5280] on the contents of Workload Identity Certificates or on the certification authorities that issue workload certificates.

3.2.1. Server Name Validation

If the WIMSE client uses a hostname to connect to the server and the server certificate contain a DNS SAN the client MUST perform standard host name validation (Section 6.3 of [RFC9525]) unless it is configured with the additional information necessary to perform alternate validation of the peer's workload identity. If the client did not perform standard host name validation then the WIMSE client SHOULD further use the workload identifier to validate the server. The host portion of the workload identifier is NOT treated as a host name as specified in section 6.4 of [RFC9525] but rather as a trust domain. The server identity is encoded in the path portion of the workload identifier in a deployment specific way. Validating the workload identity could be a simple match on the trust domain and path portions of the identifier or validation may be based on the specific details on how the identifier is constructed. The path

portion of the WIMSE identifier MUST always be considered in the scope of the trust domain. In most cases it is preferable to validate the entire workload identifier, see section 1.3 of [I-D.ietf-wimse-workload-creds] for additional implementation advice.

3.3. Client Authorization Using the Workload Identity

The server application retrieves the workload identifier from the client certificate subjectAltName, which in turn is obtained from the TLS layer. The identifier is used in authorization, accounting and auditing. For example, the full workload identifier may be matched against ACLs to authorize actions requested by the peer and the identifier may be included in log messages to associate actions to the client workload for audit purposes. A deployment may specify other authorization policies based on the specific details of how the workload identifier is constructed. The path portion of the workload identifier MUST always be considered in the scope of the trust domain. See section 1.3 of [I-D.ietf-wimse-workload-creds] on additional security implications of workload identifiers.

4. IANA Considerations

This document does not include any IANA considerations.

5. Security Considerations

This document relies on the security properties of TLS [TLS], PKIX path validation [RFC5280], and workload identity certificate validation as described in Section 4.1 of [I-D.ietf-wimse-workload-creds]. Implementations MUST validate the peer certificate chain, the applicable extended key usage, and the workload identifier according to the rules in this document before using the authenticated identity for authorization decisions.

Workload identifiers are meaningful only within the scope of their trust domain. Authorization policies MUST NOT evaluate only the path or other sub-components of a workload identifier without also considering the trust domain and the trust anchor used to validate the certificate. Failure to bind the workload identifier to the expected trust domain and configured trust anchor can allow one trust domain to impersonate workloads from another domain.

Server authentication can be based on either conventional DNS name validation or workload identity validation, depending on deployment configuration. If DNS name validation is not performed, the client MUST be configured with sufficient information to determine the expected workload identity of the server. Accepting any certificate issued by a trusted workload CA without validating that it represents the intended server workload would allow mis-issued or otherwise valid certificates for other workloads to be used for impersonation.

Client authentication is based on the workload identity certificate presented by the TLS client. A server performing mTLS authentication MUST validate the client certificate chain, the associated trust domain, and the workload identifier before using that identity for authorization, accounting, or auditing. Accepting any valid client certificate from a trusted CA without checking whether the authenticated workload is authorized for the requested action can allow unintended workloads to gain access.

Workload identity certificates are often issued to dynamic or short-lived workloads. Deployments SHOULD use certificate lifetimes that are appropriate for the workload environment and SHOULD provide timely revocation or replacement mechanisms when workload identity, authorization, or runtime state changes. Long-lived certificates increase the impact of private key compromise and stale authorization decisions.

Private keys associated with workload identity certificates MUST be protected against disclosure and unauthorized use. In particular, deployments MUST NOT share private keys across unrelated workload instances. Where possible, private keys SHOULD be generated and held in the workload runtime environment or a dedicated key protection mechanism, rather than distributed over the network.

This document specifies authentication at the TLS layer. If application traffic traverses intermediaries, gateways, service meshes, or other middleboxes that terminate and re-establish TLS, the application endpoint might not be directly authenticated to the peer workload. In such deployments, authorization decisions need to account for where TLS is terminated and whether the authenticated certificate represents the peer workload, an intermediary, or another delegated entity. Where end-to-end workload authentication context is required across such boundaries, deployments SHOULD use an application-layer WIMSE protection mechanism in addition to TLS-layer server authentication.

Client certificate authentication exposes the client workload identity to the TLS server during the handshake. Deployments should consider whether disclosure of workload identifiers to servers,

intermediaries, or logs is acceptable for their threat model. Workload identifiers included in certificates and audit records should avoid embedding unnecessary sensitive information.

Authorization decisions based on workload identity need to be made using the authenticated identity obtained from the validated certificate, not from unauthenticated application-layer metadata such as HTTP headers. Application-layer identity assertions can be useful for logging or context, but they MUST NOT override the identity established by mutual TLS unless protected and authorized by another mechanism.

6. References

6.1. Normative References

[I-D.ietf-wimse-workload-creds]
Campbell, B., Salowey, J. A., Schwenkschuster, A., Sheffer, Y., and Y. Rosomakho, "WIMSE Workload Credentials", Work in Progress, Internet-Draft, draft-ietf-wimse-workload-creds-00, 3 November 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-wimse-workload-creds-00>.

[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>.

[RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, DOI 10.17487/RFC5280, May 2008, <https://www.rfc-editor.org/rfc/rfc5280>.

[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>.

[RFC9525] Saint-Andre, P. and R. Salz, "Service Identity in TLS", RFC 9525, DOI 10.17487/RFC9525, November 2023, <https://www.rfc-editor.org/rfc/rfc9525>.

[TLS] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", Work in Progress, Internet-Draft, draft-ietf-tls-rfc8446bis-14, 13 September 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-tls-rfc8446bis-14>.

6.2. Informative References

[I-D.ietf-wimse-arch]
Salowey, J. A., Rosomakho, Y., and H. Tschofenig, "Workload Identity in a Multi System Environment (WIMSE) Architecture", Work in Progress, Internet-Draft, draft-ietf-wimse-arch-07, 2 March 2026, <https://datatracker.ietf.org/doc/html/draft-ietf-wimse-arch-07>.

Appendix A. Document History

// RFC Editor: please remove before publication.

A.1. draft-ietf-wimse-mutual-tls-01

* Added security considerations

A.2. draft-ietf-wimse-mutual-tls-00

* Initial version, extracted from the draft-ietf-wimse-s2s-protocol-07 S2s draft with minimal edits.

* added security consideration for Server Name Validation

Acknowledgments

We thank Yaron Sheffer, Arndt Schwenkschuster, Brian Campbell, and Daniel Feldman for their contributions to earlier versions of this document.

Authors' Addresses

Joe Salowey
CyberArk
Email: joe@salowey.net

Yaroslav Rosomakho
Zscaler
Email: yaroslavros@gmail.com