

Additional Hash Algorithms for OAuth 2.0

draft-skokan-oauth-additional-hashes

Filip Skokan & Aaron Parecki, Okta

IETF 125 Shenzhen, March 2026

The Problem

- **Three OAuth 2.0 mechanisms hard-code SHA-256** with no alternative or extension point:
 - **PKCE** (RFC 7636): S256 code challenge method
 - **Mutual-TLS** (RFC 8705): x5t#S256 certificate thumbprint in cnf confirmation claim
 - **DPoP** (RFC 9449): jkt key thumbprint in cnf confirmation claim, ath access token hash in DPoP Proof JWT
- Security policies like **CNSA 2.0 prohibit SHA-256**, requiring SHA-384 or SHA-512
- **Result:** These OAuth mechanisms cannot be used in policy-constrained environments

Purpose & Scope

- This draft defines **SHA-512 alternatives** only for deployments where SHA-256 is prohibited
- **SHA-256 is not deprecated**, it remains the recommended default
- Deployments not subject to such policies **SHOULD NOT** use these new methods
- Negotiation and metadata mechanisms enable **coexistence** and **backward compatibility** should SHA-256 ever need to be deprecated broadly

New SHA-512 Methods

Mechanism	Existing (SHA-256)	New (SHA-512)	Where
PKCE	S256	S512	
mTLS	x5t#S256	x5t#S512	AT cnf claim
DPoP thumbprint	jkt	jkt#S512	AT cnf claim
DPoP token hash	ath	ath#S512	DPoP Proof JWT
DPoP authz code binding	<i>(implicit S256)</i>	dpop_jkt_method parameter	Authorization request

Discovery & Negotiation

AS Metadata (RFC 8414): code_challenge_methods_supported (existing),
dpop_jkt_methods_supported (new)

RS Metadata (RFC 9728): mtls_confirmation_methods_supported (new),
dpop_confirmation_methods_supported (new),
dpop_access_token_hash_methods_supported (new)

DPoP WWW-Authenticate challenge: ath_methods parameter (new)

All parameters **default to SHA-256 when absent**

Call for Reviews

Read the draft:

<https://datatracker.ietf.org/doc/draft-skokan-oauth-additional-hashes/>

GitHub:

<https://github.com/panva/draft-oauth-additional-hashes>

Feedback welcome, please review and comment!