

256t.org

A Public Specification for Content Addressable Storage

A standard defining a domain where the URL path uniquely defines the content it serves.

`https://256t.org/AgAA...{94-chars}...zQ=`

The Deterministic Map

- **Match:** Content served.
- **No Match:** Definitive 404.

The last element of the URL path (up to 94 characters) uniquely defines the payload.

Addressable Capacity: 256 Terabytes

Prefix Length: 8 characters (Base64)

Bit Depth: 8 chars × 6 bits = 48 bits

Address Space: 2^{48}

Expansion: $2^{40} \times 2^8$

Breakdown: $2^{10(K)} \times 2^{10(M)} \times 2^{10(G)} \times 2^{10(T)} \times 2^8$

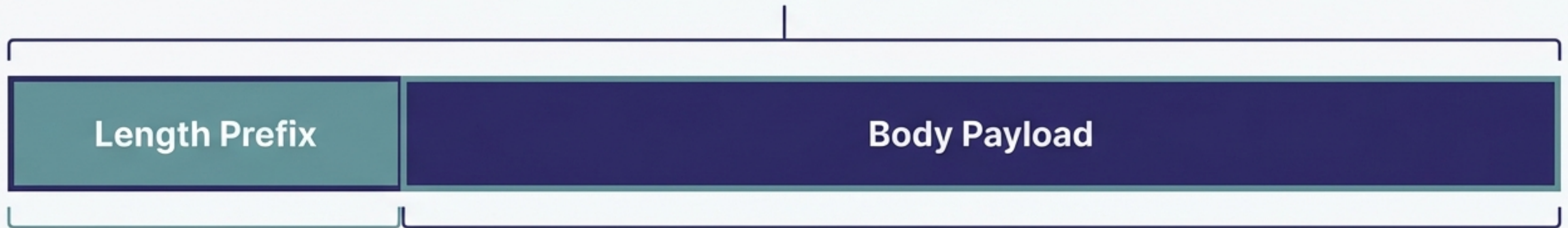
= 256 Terabytes



A massive namespace for unique content addressing.

Anatomy of the Path

Total Max Length: 94 Characters



8 Characters

Purpose: Defines Content Size

Purpose: Defines Content Size

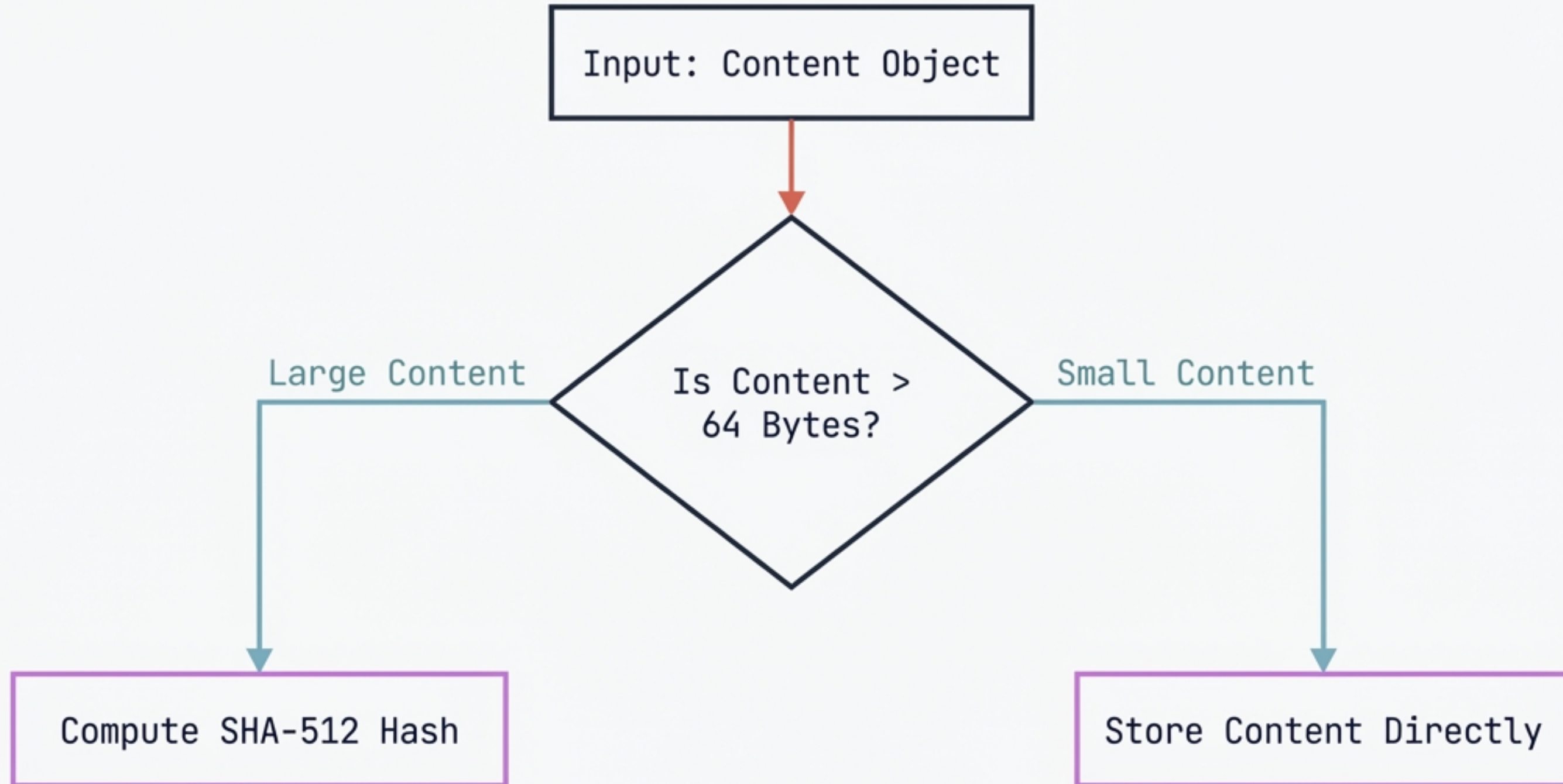
86 Characters

Purpose: Contains Hash OR Raw Content

Purpose: Contains Hash OR Raw Content

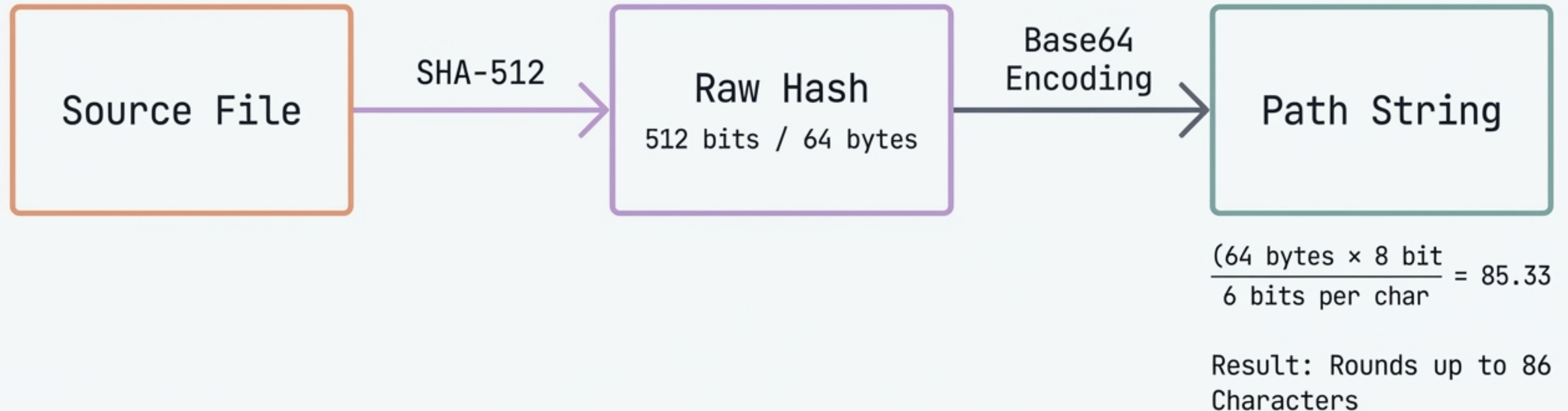
Encoding: Base64URL (RFC 4648)

Logic Flow: The Storage Decision



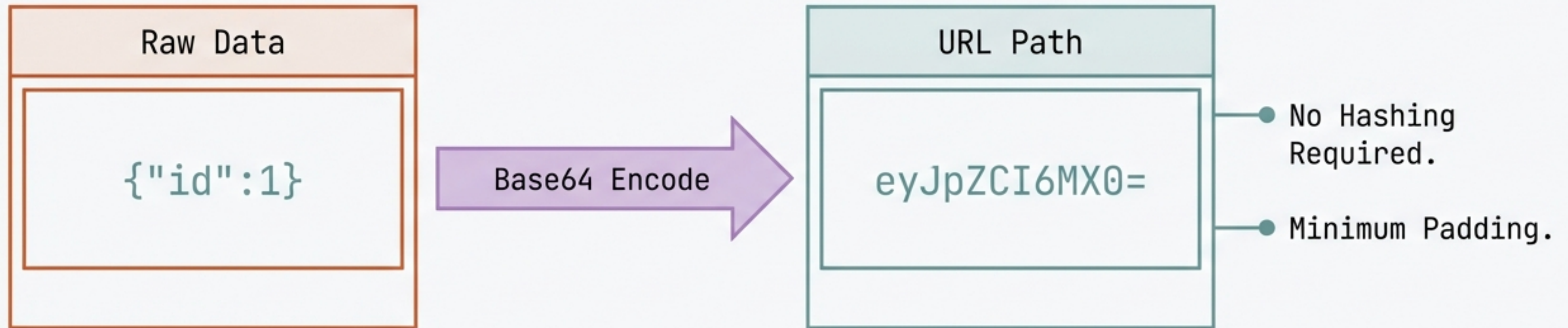
Optimization: Don't hash what you can carry.

Scenario A: Large Content (> 64 Bytes)



The path is a cryptographic fingerprint.

Scenario B: Small Content (≤ 64 Bytes)



Efficiency: The content IS the identifier.
Zero database lookups required.

The Encoding Standard: Base64URL

RFC 4648 Compliant

Encoding Type	Characters	Status
Standard	Includes '+' and '/'	⚠ Unsafe for URLs (Requires escaping)
Base64URL	Replaces with '-' and '_'	✓ Web Safe

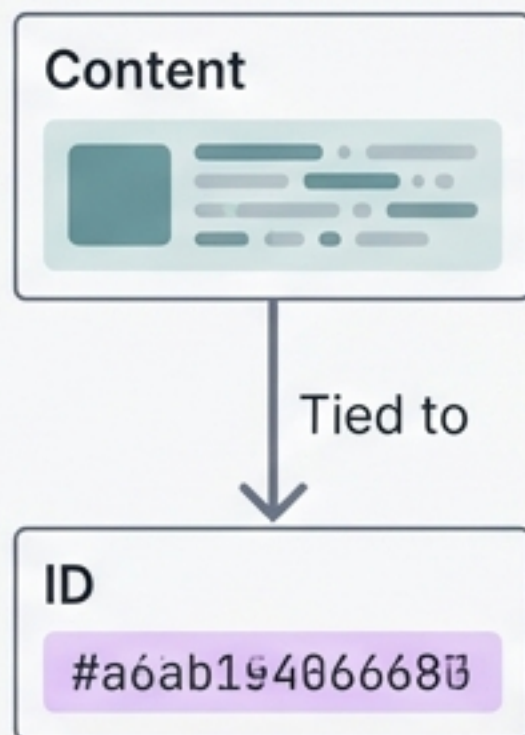
- ✓ Filename safe
- ✓ Command-line argument safe
- ✓ No percent-encoding needed (e.g. %20)

Immutability & Caching

Theory

Inter Tight Semibold

Content is tied to its ID. If the content changes, the ID changes.



Practice - HTTP Headers

Inter Tight Semibold

```
1 Cache-Control: public, max-age=31536000, immutable
2
3 Status: 200 ✓ OK (Match)
4
5 Status: 404 × Not Found (Mismatch)
```

Network Effect: ETags are redundant. 404s are definitive.
Network Effect: ETags are redundant. 404s are definitive.

Handling Collisions

Accidental

Pure chance.
Risk: Ignorable.



Adversarial

Intentional
manipulation.



Existing

CID produced from
different content.



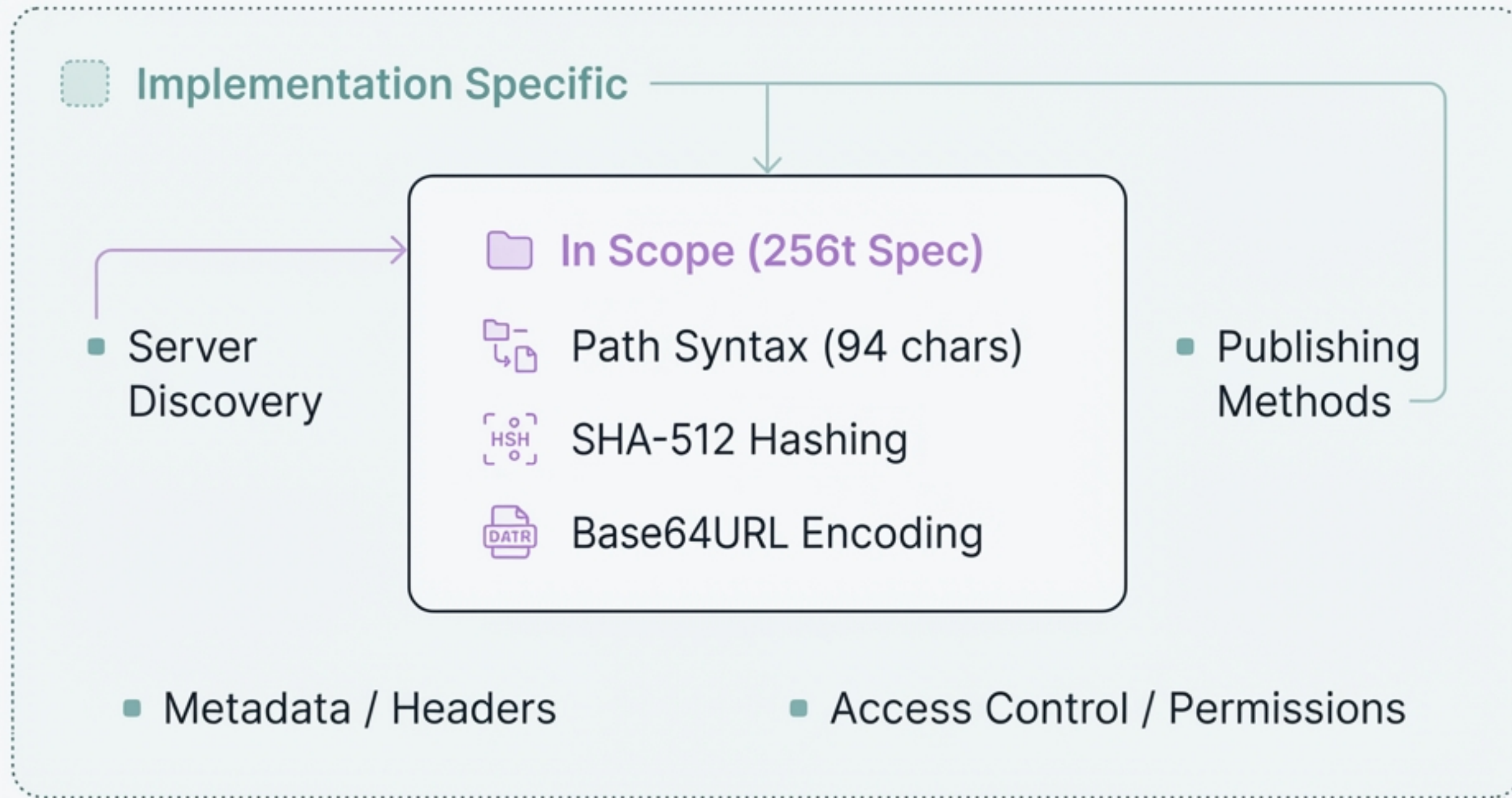
The Solution

Trust & Verify.



Mitigation Strategy: Always calculate the hash of received content. It is easier to lie than to engineer a SHA-512 collision.

Scope of Specification



Developer Tooling

Hash Calculator

Input Text

SHA-512 Output ↓

...f7a9c8b2...

Calculate

Instant text-to-SHA512 conversion.

JetBrains Mono

CID Checker

File icon → CID bafy... ✓

JS Elm

Verify files against ID (JS & Elm).

JetBrains Mono

File Analysis

Drop File Here

Progress bar

Drag-and-drop local analysis.

JetBrains Mono

Downloader

Servers → File icon

Verify Fetch

Fetch & verify from compatible servers.

JetBrains Mono

Implementation Ecosystem

System	Web	Scripting	Enterprise	Functional
Rust	TypeScript	Python	Java	Haskell
Go	JavaScript	Ruby	C#	Clojure
C	Elm	Bash	Kotlin	OCaml
C++	PureScript	Perl Lua		F#

Language Agnostic. Portable across any stack supporting SHA-512 & Base64.

Language Agnostic. Portable across any stack supporting SHA-512 & Base64.

A Simple Primitive for the Complex Web.

256t provides the storage mechanism.
You build the discovery and permission layers.

<https://github.com/curtcox/256t.org>

Read the spec. View examples. Contribute.