

Resumable Uploads

draft-ietf-httpbis-resumable-upload

IETF 115, 11 Nov 2022

Marius Kleidl

Resumable Uploads

- Example implementation: <https://github.com/tus/draft-example>
- Upload flow:
 - Upload Creation Procedure (POST etc)
 - Upload Appending Procedure (PATCH)
 - Offset Retrieving Procedure (HEAD)
 - Upload Cancellation Procedure (DELETE)

Major open issues

- Server-generated URL vs client-generated token ([#2239](#))
- Retry-able upload creations using Idempotency-Key ([#2293](#))
- Request identification using Prefer header ([#2295](#))
- Conflicts with Expect header ([#2243](#))

Server-generated URL vs client-generated token ([#2239](#))

- Draft -00 uses client-generated token
 - Upload-Token header is used to identify the upload resource
- Proposal to use server-generated upload URLs ([#2292](#))
 - Server responds with upload URL in 1xx and 2xx response
 - Achieves same benefits as client-generated tokens (retryability, single request uploads)
- Discussion: Switch to server-generated URLs?

Use of Idempotency-Key ([#2293](#))

- Client can use Idempotency-Key to retry Upload Creation Procedure
 - Enables retryable, single request uploads for small files
 - Only if supported by server
- Discussion: Should Idempotency-Key be used for this?
 - If so, how to reference the Idempotency-Key draft?

Use of Prefer header ([#2295](#))

- Use `Prefer: resumable-upload` header to identify the Upload Creation Procedure
- Discussion: Is this a good way of identification?

Conflicts with Expect header ([#2243](#))

- Server can send informational response (104 Resumption Supported)
- Discussion: Is a conflict with `Expect: 100-continue` possible (order of 100 and 104 response)?

All open issues

- **Server-generated URL vs client-generated token ([#2239](#))**
 - **Retry-able upload creations using Idempotency-Key ([#2293](#))**
 - **Request identification using Prefer header ([#2295](#))**
 - **Conflicts with Expect header ([#2243](#))**
-
- Non-cachable HEAD responses ([#2294](#))
 - Upload progress using informational responses ([#2291](#))
 - Prioritization of uploads ([#2242](#))
 - Integration into HTML forms ([#2247](#))