Chunked Oblivious HTTP Messages

draft-ietf-chunked-ohttp-01

Tommy Pauly & Martin Thomson
OHAI
July 2024, Vancouver
Chunked OHTTP

Chunked OHTTP allows encrypting and decrypting requests and responses in separate chunks

Allows the use of Binary HTTP's "indeterminate" mode

Takes advantage of HPKE's support for multiple messages

Still is a **single** HTTP request-and-response transaction
Updated in -01

Fixed Issue #9 around maximum number of chunks

Response chunk counter must not exceed $2^Nn$, since this would cause nonce reuse

Still need to fill in test vectors and protocol formal analysis
Incremental forwarding

Issue #19

Implementation and deployment testing found that most common CDN implementations made chunked OHTTP ineffective.

In practice, POSTs are often buffered and only forwarded by the intermediaries when the content is complete.
Incremental forwarding
Issue #19

RFC 9110, Section 7.6:

An HTTP message can be parsed as a stream for incremental processing or forwarding downstream. However, senders and recipients cannot rely on incremental delivery of partial messages, since some implementations will buffer or delay message forwarding for the sake of network efficiency, security checks, or content transformations.
Incremental forwarding
Issue #19

Options

**Deployment fix:** Intermediaries that are used for chunked OHTTP will update to not buffer this content

**Design fix:** Replace POST with a new form of extended CONNECT (but this loses the advantages of making OHTTP easy to deploy with POST)

**Negotiation fix:** Allow clients and servers to agree about buffering for POST, etc
Incremental forwarding

Issue #19

Proposal for "negotiation fix" from Kauho!

New header field clients send in requests for which the content should be streamed instead of buffered

Request-Streaming = 1

Thoughts? Bikesheds?

Where should this work be done?
Next steps

Finish updates based on feedback at 119
- Max chunk sizes
- Configuration recommendations
Handle incremental forwarding
More implementations/testing
- Already multiple client/relay/server implementations being used