Chunked Oblivious HTTP Messages

draft-ietf-chunked-ohttp-00

Tommy Pauly & Martin Thomson
OHAI
March 2024, Brisbane

Agenda

Recap of protocol

Open issues

Next steps

Chunked OHTTP

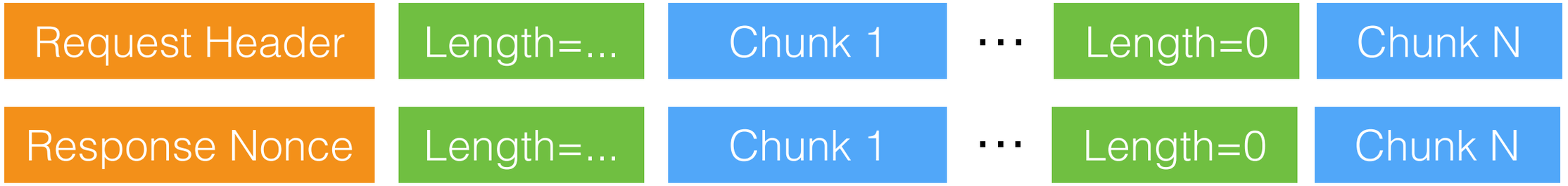
Chunked OHTTP allows encrypting and decrypting requests and responses in separate chunks

- 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

How to chunk OHTTP

Add a varint "length" field before each chunk

Final chunk is indicated by length=0, and extends to the end of the outer stream



Prevents reordering of chunks and truncation by removing chunks

New media types

`message/ohttp-chunked-req`

`message/ohttp-chunked-res`

Negotiating use

Issue #5

How can the client know the gateway supports chunked OHTTP?

A. Out-of-band configuration

B. OHTTP key configuration

   - Would this need a different media type than "application/ohttp-keys", such as "application/ohttp-chunked-keys"?
   - Client could issue a GET to the Oblivious Gateway with the new type to ask it if it supports chunked

C. Optimistically attempting requests with the chunked type

   - Risks inconsistent or client-targeting behavior

D. Something else?

Negotiating use

Issue #5

If support for chunking is part of the gateway configuration, it should be subject to consistency checks

Groups of clients should consistently use either chunked or non-chunked, but not a mix

Maximum chunk sizes

Issue #7

Do we need to restrict chunk sizes to a maximum?

- Maximum chunk sizes limit the amount of memory that can be required to process a varint-based length
- If negotiated, this value will need to be checked for consistency
- OHTTP without chunking doesn't have a maximum size, so is this not needed?
- Gateways and client can have a reasonable non-negotiated maximum that they automatically enforce

Next steps

Add protocol formal analysis

Add test vectors

Expand privacy discussion (timing leaks when using 100-continue, etc)