# Chunked Oblivious HTTP Messages

*draft-ietf-chunked-ohttp-03*

Tommy Pauly & Martin Thomson
OHAI
Virtual Interim 2024

# 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

# Updates in -03

Media type usage

Require that if requests used the chunked media type, responses MUST also use the chunked media type

Explain that clients SHOULD commit to one media type or another, and not fall back between them

# Updates in -03

## Maximum chunk size

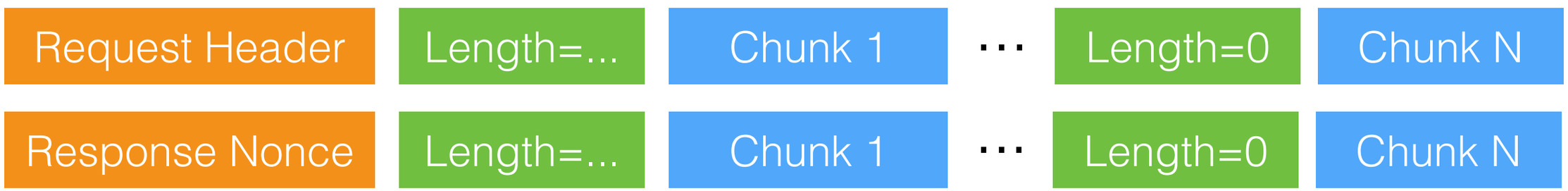
```
Implementations MUST support receiving chunks
that contain 2^14 (16384) octets of data prior
to encapsulation.  Senders of chunks SHOULD
limit their chunks to this size, unless they
are aware of support for larger sizes by the
receiving party.
```

Point to confirm: this is applying to the plaintext size inside the chunk, to make it easier to limit the size on sending. Should it be the size after encoding?

# Updates in -03

Pseudocode fix

Fixed pseudocode example for final chunk, to make it clear that the final chunk is prefixed by a "0" length



# Updates in -03

Security considerations

Explain interactivity in more detail

- Any case where the client request content or timing is influenced by the response
- Cases with single request chunks, or request chunks that finish before the response, are not interactive
- Interactive behavior is a choice of the client, and clients need to only do so if the risks of identification are acceptable (if they need `100-continue`, etc)
- Note that interactivity does not reduce replay risk, as it might in other protocols where there is PFS

# What's left

Test vectors

Formal analysis

Media type request

Track dependency on Incremental header field

Adoption call in HTTPBIS is out