# RateLimit Headers

Communicate service status

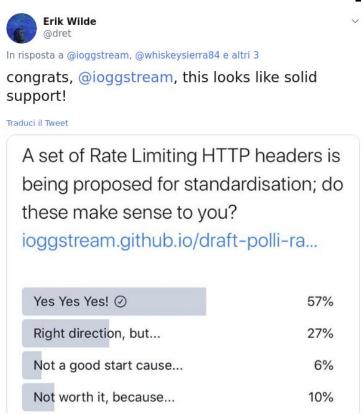
HTTPAPI-WG @ IETF-109

draft-rpolli-ratelimit-headers
[see the specifications]

#### RateLimit HTTP Fields - Goals

- communicate service limits, so clients can stop before being throttled out
- align all the \*already existing\* ratelimit headers and stop headers proliferation
- express multiple RateLimit policies

### Who wants it & Implementers



### Configurable in:

- Red Hat 3scale
- Kong
- Envoy
- Azure API Gateway

### Supported by:

- Italy
- The Netherlands

### STOP headers proliferation

X-RateLimit-UserLimit: 1231513

X-RateLimit-UserRemaining

X-Rate-Limit-Limit: name=rate-limit-1,1000

x-custom-retry-after-ms

x-ratelimit-minute: 100

x-rate-limit-hour: 1000

X-RateLimit-Remaining-month

X-RateLimit-Retry-After: 11529485261

X-Rate-Limit-Reset: Wed, 21 Oct 2015 07:28:00 GMT

RateLimit-Limit: #quota-units

RateLimit-Remaining: #quota-units

RateLimit-Reset: #delta-seconds

... and many more!

### Example with multiple quotas

mandatory part

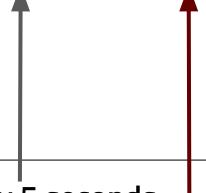
optional parts with policy details and comments

RateLimit-Limit: 10

RateLimit-Remaining: 6

RateLimit-Reset: 3

, 10;w=5 , 80;w=60;comment="bar"



10 units every 5 seconds | AND 80 units every 60 seconds

#### **Technical choices**

- #60 support only delta-seconds (no ntp skew & adjustment issues) like Retry-After
- #49 quota expressed in units, may or may not be requests support multiple quota policies and comments
- flexible semantics to express dynamic policies, sliding windows and concurrency limits
- don't mention infrastructural concepts like connections

## **Open Issues Needing Input**

- #86 Refine normative language WRT intermediaries
- #35 Use Structured-Headers
- #84 Define a throttling scope, related to Retry-After
- #42 Define header dependencies ?
- #41 Upper bound for RateLimit-Reset ?

### **FAQ**

### Q: Are we inventing a new service management model?

A: No. We just standardize headers semantic for the many who \*already\* use this pattern.

#### Q: Why don't use timestamps for RateLimit-Reset?

A: Timestamps \*require\* NTP on both sides. NTP in the real world is hard (skew, adjust, IoT, ...). We like Retry-After too;)

#### Thanks!

Roberto Polli - robipolli@gmail.com

Alex Martinez - amr@redhat.com

# **Backup slides**

### Example...after 40 seconds

mandatory part

optional comment parts with policy details

RateLimit-Limit: 80

RateLimit-Remaining: 0

RateLimit-Reset: 20

, 10;w=5 , 80;w=60;foo="bar"

^--- now use this

After 40 seconds, client consumed 80 units. The enforced quota is the second one.

# Why proliferation is bad?

Currently every API gateway implements custom ratelimit headers

Clients consuming APIs behind different gateways have to support different ratelimit headers.

The reality is that they ignore them