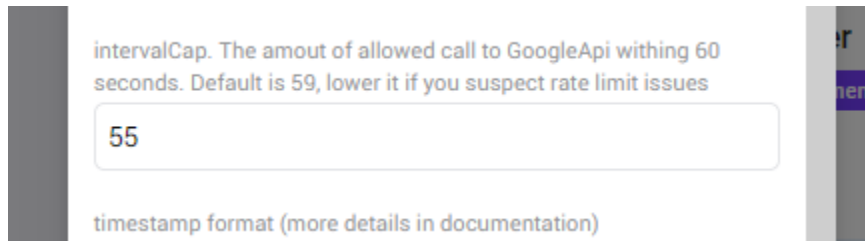


## 12. intervalCap

This option is on configuration page. If not set, default value is 59. What it means is that it will limit the amount of calls to GoogleApi within 60 seconds. If the number of calls is above the intervalCap value within 60 seconds, it will be queued and get a delayed invocation. It should always be executed, but with some delay to avoid rate limit situations at Google.

Just play with it. The App needs to be restarted after updating the value.

This should be used when utilizing [11. Fire and forget](#)

A screenshot of a configuration page for 'intervalCap'. The page has a dark grey header and footer. The main content area is white. At the top, there is a description: 'intervalCap. The amount of allowed call to GoogleApi within 60 seconds. Default is 59, lower it if you suspect rate limit issues'. Below this is a text input field containing the number '55'. At the bottom, there is a label 'timestamp format (more details in documentation)'.

### How does it work

Every request to Google is put into a queue, where only one request at the time is executed. This is important in a [fire and forget](#) configuration. For some calls the order is important. And this will ensure that. Next is that the queue order will try to count request within a specific timeslot. This timeslot is fixed at 60 seconds, to match Google API free tier.

Next the configurable setting of intervalCap(acity) is adjustable for each user. The default value of 59 is to match the Google API free tier. If you experience rate limit, it can be because the algorithm at Google is different than the Easy Logger Queue, so in that case you can lower it and see if it works better for you.

How can you tell? If your flows are slow, or seems to hang, or misses data, it may be because of the rate limit, http code 429. Sending a diag to developer might help to sport if that is a possible reason.

*Happy logging !*