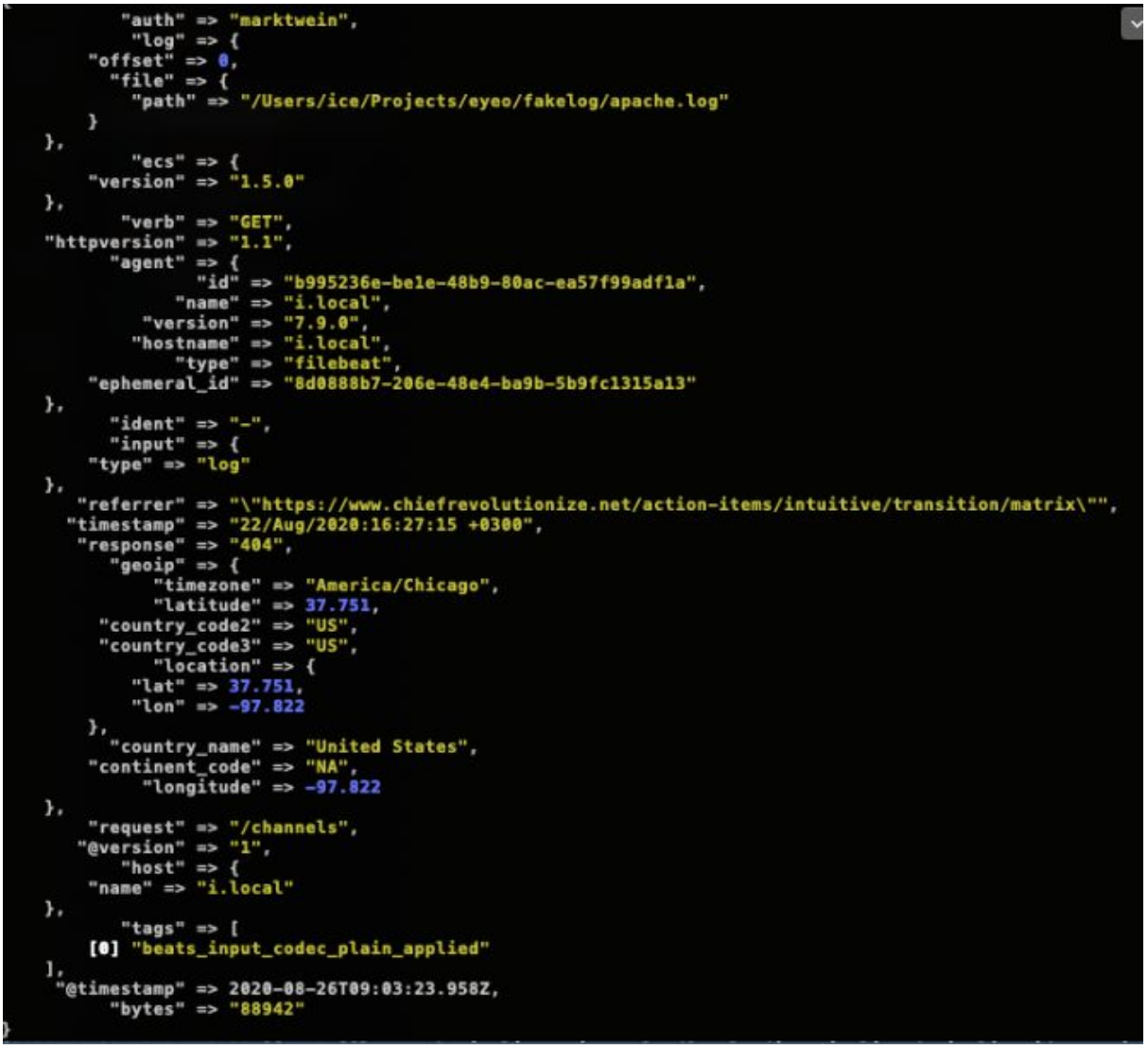
There is an almost perfect ELK solution for this task using **logstash**.
1. Excellent processing speed
2. Possibility of configuration, including file masks
3. Excellent customizable filters
4. Ability to distribute components to different servers (logstash, elastic search, filebeat)

Speaking of the configuration in the test description of the logs on the Apache server, they are default.
The **grok** plugin is great for data representation, and it already has a built-in mask for Apache logs.
To search geolocation of IP:
You don't need to use a third party because logstash provides a built-in solution via using the **geoip** plugin.
Any extra data will be hidden -> using mutate. We can also configure the data blur through **gsub**, instead of hiding them
Sample of output data after filebeat + logstash solution.



Python
Most likely the task was set to test the coding skills, so I decided to write this solution.

We need to solve several problems at once.
I/O bounded task:
0. Reading a file
1. Waiting for geoip results
2. Writing data to the database (we don't care of the order of data because we cant sort via timestamp)

CPU:
1. Processing multiple files

Solution for i / o tasks:
Asynchronous coding will be applied, as an alternative, you can consider using the thread library or use one of the distributed software broker (such as AMQP, Kafka)
Solution for CPU tasks:
we can use multiprocessing for the list of files
we can run the script many times using **cron** job tasks

Sample of MongoDB data after using my script:

_id	ip	ident	auth	timestamp	method	request	httpversion	status_code	bytes_	referrer	extra	timezone	latitude	lon
5f465145db160de96e6baf08	173.97.**	-	marktwein	22/Aug/2020:16:27:15 +0300	GET	/channels	HTTP/1.1	404	8942	https://www.chiefrevolutionize.net/action-items/intuitive/transition/matrix	Opera/9.24 (Macintosh; PPC Mac OS X 10_6_4; en-US) Presto/2.11.257 Version/13.00	America/Chicago	37.751	97.822