

1. api_http_request_duration_seconds_sum:

- This metric represents the sum of durations for all HTTP requests falling within the specified criteria.
- {endpoint="/metrics", methods="GET", status="200", le="0.005"}: These are labels associated with the metric, providing context about the requests being measured.
 - endpoint="/metrics": Indicates that the endpoint being accessed is "/metrics".
 - methods="GET": Indicates that the HTTP method used is a GET request.
 - status="200": Indicates that the HTTP status code returned is 200, which typically means the request was successful.
 - le="0.005": Indicates the upper bound of the time range for this bucket. In this case, it's 0.005 seconds, or 5 milliseconds.
- 8: This is the value associated with the metric. It represents the total sum of durations (in seconds) for all requests that fall within the specified duration bucket.

2. api_http_request_duration_seconds_count:

- This metric represents the count of HTTP requests falling within the specified criteria.
- {endpoint="/metrics", methods="GET", status="200", le="0.005"}: These are the same labels as in the previous metric.
- 8: This is the value associated with the metric. It represents the total count of requests that fall within the specified duration bucket.

3. Api_http_request_duration_seconds_bucket:

This is the name of the metric, indicating that it tracks the duration of HTTP requests.

{endpoint="/metrics", methods="GET", status="200", le="0.005"}: These are labels associated with the metric, providing additional context about the requests being measured.

- endpoint="/metrics": Indicates that the endpoint being accessed is "/metrics".
- methods="GET": Indicates that the HTTP method used is a GET request.
- status="200": Indicates that the HTTP status code returned is 200, which typically means the request was successful.
- le="0.005": Indicates the upper bound of the time range for this bucket. In this case, it's 0.005 seconds, or 5 milliseconds.

8: This is the value associated with the metric. It represents the number of HTTP requests that fall within the specified duration bucket (i.e., the number of requests that were processed within 5 milliseconds).

4. CPU Usage (9):
 - This indicates the percentage of CPU utilization by the application. A value of 9 suggests that the application is currently utilizing 9% of the available CPU resources.
5. Memory Usage (22):
 - This indicates the percentage of memory utilized by the application. A value of 22 suggests that the application is currently using 22% of the available memory.
6. Process CPU Seconds Total (44):
 - This metric represents the total CPU time consumed by the process since it started. It typically indicates the cumulative amount of CPU time spent executing the application's code and handling system tasks.
7. Process Max File Descriptors (1048432):
 - This metric indicates the maximum number of file descriptors (file handles) that the process can open. File descriptors are used to access files and other input/output resources.
8. Process Open File Descriptors (2152):
 - This metric indicates the current number of open file descriptors by the process. It provides insight into the process's file I/O activity and resource utilization.
9. Process Resident Memory Bytes (34534252):
 - This metric indicates the amount of physical memory (RAM) currently allocated to the process. It represents the memory actually in use by the process at a specific moment.
10. Process Start Time Seconds (2385252572):
 - This metric represents the Unix timestamp indicating the time when the process started. It helps track the duration since the process began execution.
11. Process Threads (433):
 - This metric indicates the number of threads spawned by the process. Threads are units of execution within the process, and this metric reflects the concurrency level of the application.
12. Process Virtual Memory Bytes (23428485478):
 - This metric indicates the total virtual memory allocated to the process. Virtual memory includes both physical RAM and swap space on disk, and it represents the address space available to the process.