

GCP Logs & Traces

Context : <https://github.com/googleapis/google-cloud-dotnet/issues/9868>

Summary :

When using `AddGoogleDiagnosticsForAspNetCore` with `CloudRun` and `BufferOptions.NoBuffer()`, it causes a big latency

However, with `CloudRun`, this settings seems required

Process :

Create a simple app where we can just change log and trace buffer options with env var
Deploy on `CloudRun`

Results :

NoBuffer / NoBuffer

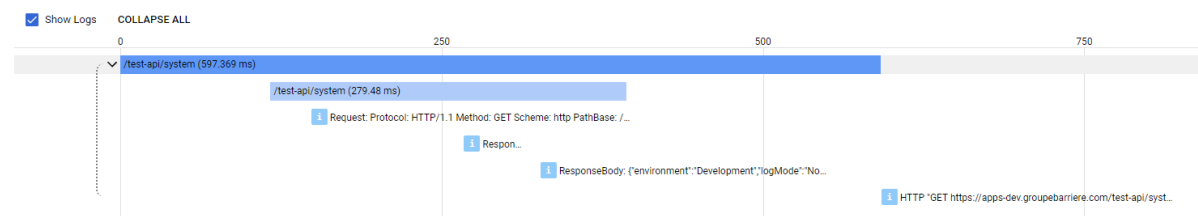
2 requests sent from browser

2 traces correctly working

Huge latency

>	i	2023-03-03 10:14:29.507 CET	GET	200	823 B	591 ms	Chrome 110.0...	https://apps-dev.groupebarriere.com/test-api/system
>	i	2023-03-03 10:14:07.631 CET	GET	200	824 B	519 ms	Chrome 110.0...	https://apps-dev.groupebarriere.com/test-api/system

Trace details



After each log, we have “white” => this is the time needed to upload log/trace

Around 100ms for each “white”

Timed0 / Timed0 (which means 10ms timedbuffer for both logs and traces)

3 requests sent from browser (ignoring 1st request response time because of cold start)

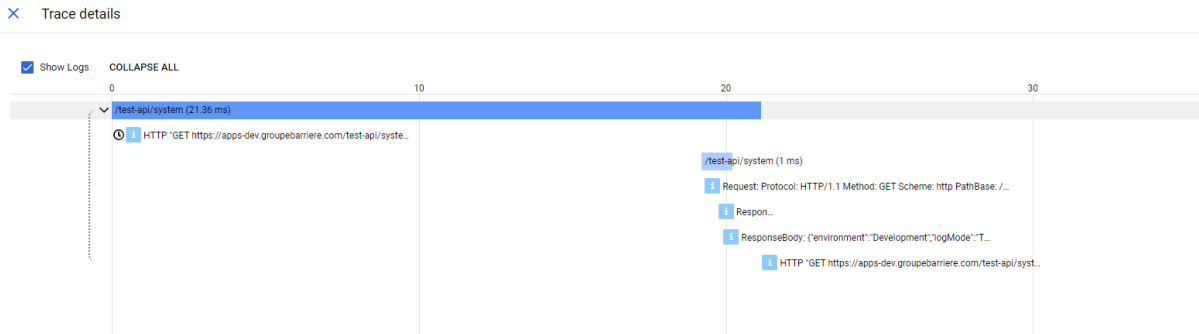
2 traces correctly working, the 1st and the 3rd (2nd one only has logs but no trace ...)

No Latency

>	i	2023-03-03 10:26:54.055 CET	GET	200	819 B	19 ms	Chrome 110.0...	https://apps-dev.groupebarriere.com/test-api/system
>	i	2023-03-03 10:26:51.100 CET	GET	200	817 B	5 ms	Chrome 110.0...	https://apps-dev.groupebarriere.com/test-api/system
>	i	2023-03-03 10:26:44.482 CET	GET	200	877 B	3 s	Chrome 110.0...	https://apps-dev.groupebarriere.com/test-api/system
>	i	2023-03-03 10:26:54.055 CET	GET	200	819 B	19 ms	Chrome 110.0...	https://apps-dev.groupebarriere.com/test-api/system
>	i	2023-03-03 10:26:51.100 CET	GET	200	817 B	5 ms	Chrome 110.0...	https://apps-dev.groupebarriere.com/test-api/system
>	i	2023-03-03 10:26:44.482 CET	GET	200	877 B			ps://apps-dev.groupebarriere.com/test-api/system

Unable to explain what is going on here : if the 3rd request correctly has a trace, why the 2nd one does not

=> is this a thread throttling by CloudRun ? if yes, why the 3rd request correctly sent its logs / traces (should have also picked the previous trace, it is a buffer common to all requests ...)



The “white” area is the load balancer part, but then we can see that there are no more latency between log statements

Timed0 / NoBuffer (which means 10ms timedbuffer for logs and no buffer for traces)
 3 requests sent from browser (ignoring 1st request response time because of cold start)
 0 trace working

>	i	2023-03-03 10:37:41.334 CET	GET	200	817 B	335 ms	Chrome 110.0...	https://apps-dev.groupebarriere.com/test-api/system
>	i	2023-03-03 10:37:39.939 CET	GET	200	820 B	301 ms	Chrome 110.0...	https://apps-dev.groupebarriere.com/test-api/system
>	i	2023-03-03 10:37:36.708 CET	GET	200	880 B	3 s	Chrome 110.0...	https://apps-dev.groupebarriere.com/test-api/system

=> this was the recommendation from Google team to be sure to have traces, it does not work ...

Sized2 / Sized1 (which means 64Ko buffer for logs, 4Ko buffer for traces)
 3 requests sent from browser (ignoring 1st request response time because of cold start)
 0 trace working

Incomplete logs (i.e even after cloud run stops the service)

>	i	2023-03-03 10:42:04.331 CET	GET	200	818 B	4 ms	Chrome 110.0...	https://apps-dev.groupebarriere.com/test-api/system
>	i	2023-03-03 10:42:02.719 CET	GET	200	818 B	6 ms	Chrome 110.0...	https://apps-dev.groupebarriere.com/test-api/system
>	i	2023-03-03 10:41:59.775 CET	GET	200	878 B	2 s	Chrome 110.0...	https://apps-dev.groupebarriere.com/test-api/system

Logs in cloud run showing “Hosting stopped”

>	i	2023-03-03 10:42:04.331 CET	GET	200	818 B	4 ms	Chrome 110	https://apps-dev.groupebarriere.com/test-api/system
>	i	2023-03-03 10:42:45.257 CET	Application is shutting down...					
>	⚠	2023-03-03 10:42:45.258 CET	Hosting stopping					
>	⚠	2023-03-03 10:42:45.264 CET	Hosting stopped					