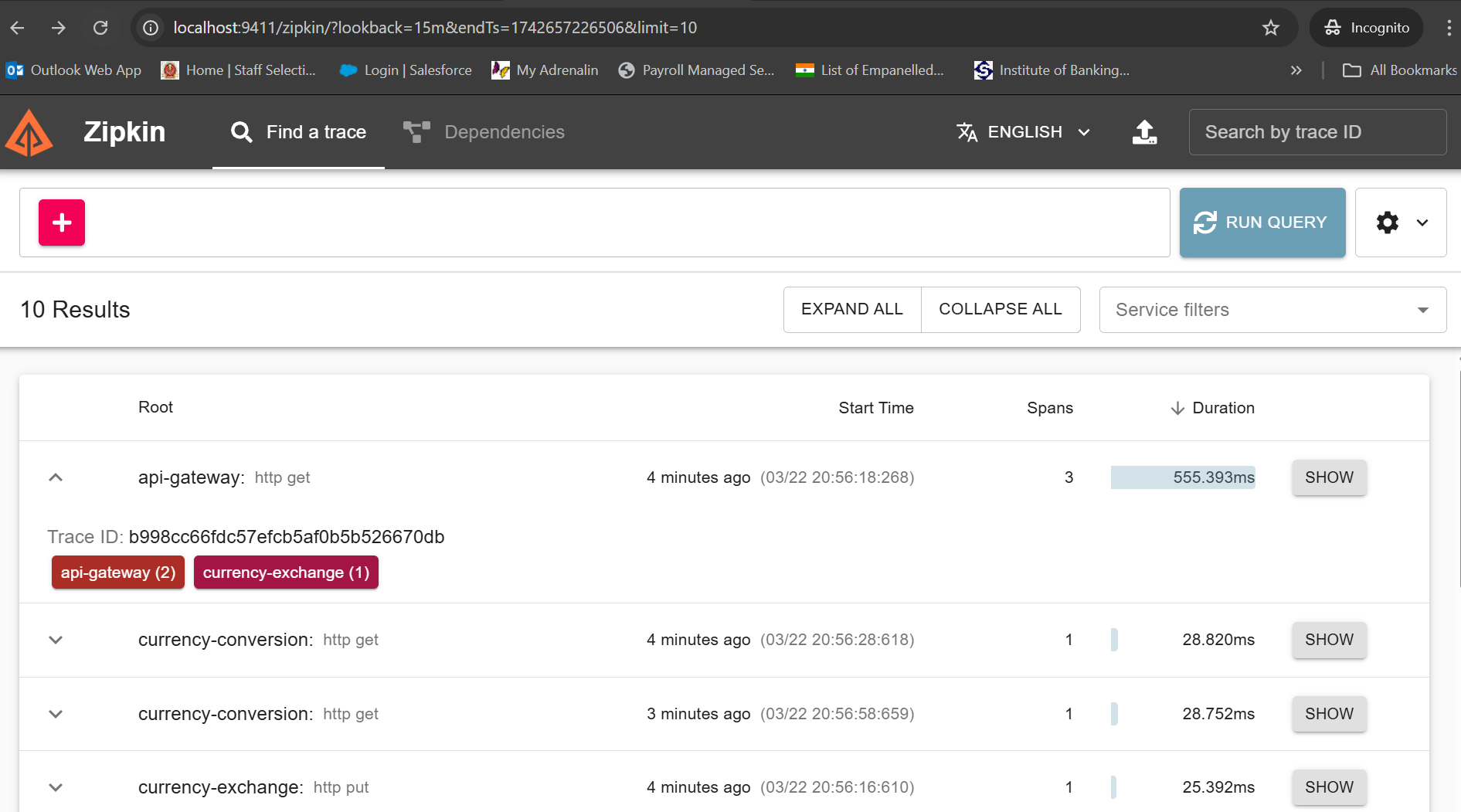
Zipkin is a distributed tracing system. It helps gather timing data needed to troubleshoot latency problems in service architectures. Features include both the collection and lookup of this data.

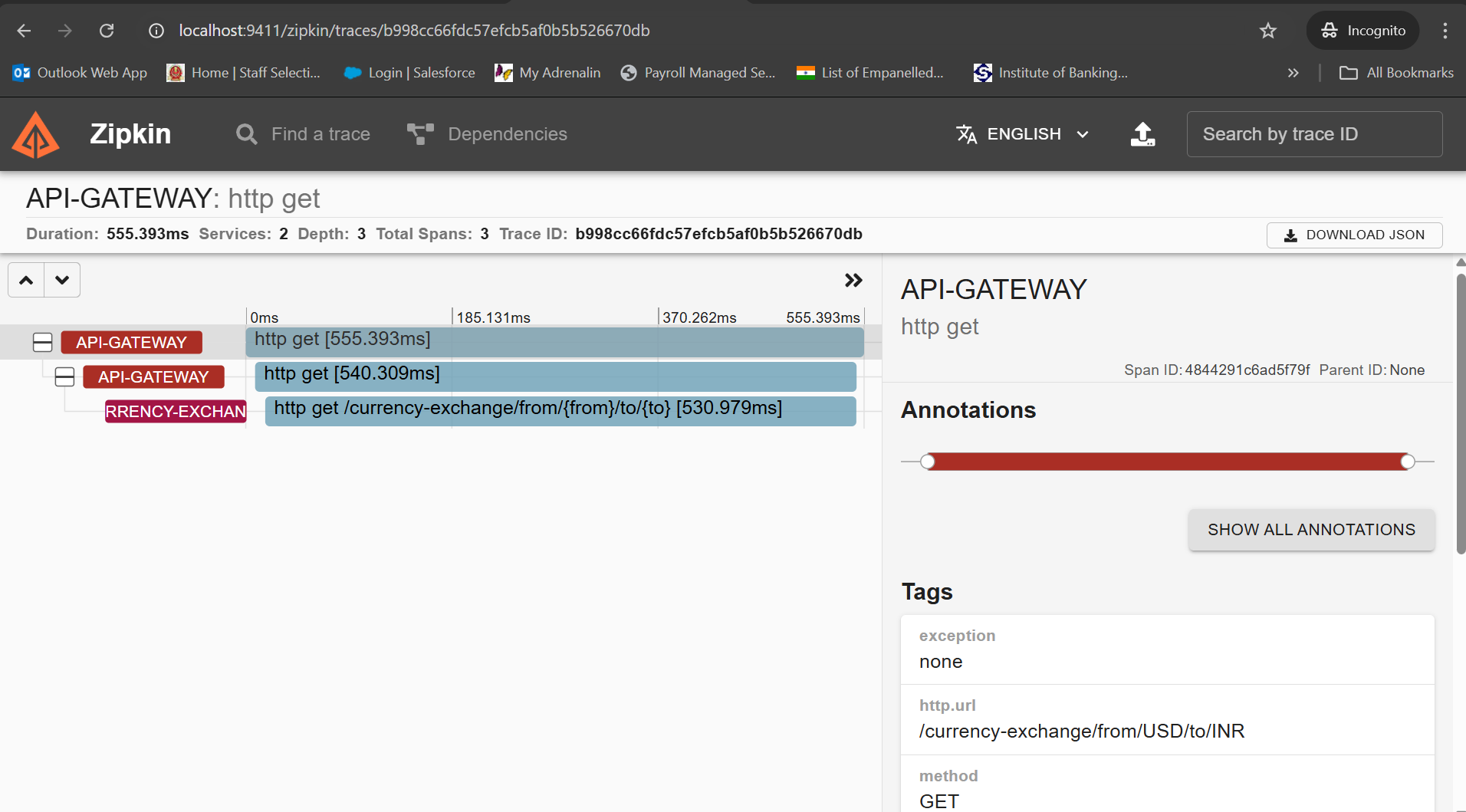
If you have a trace ID in a log file, you can jump directly to it. Otherwise, you can query based on attributes such as service, operation name, tags and duration. Some interesting data will be summarized for you, such as the percentage of time spent in a service, and whether or not operations failed. Tracing system basically used to find that how much time a request is spending in each microservice and the tracing details are stored in a database which can be check in the Zipkin’s UI (http://localhost:9411/zipkin/)

Here we will integrate the API Gateway, Currency Exchange Service and Currency Conversion Service with the Zipkin server so that request tracing details from each service are communicated to Zipkin server and then the Zipkin server can store these details in a database connected to it.

In Spring Boot-2 we use Sleuth, Brave & Zipinks to implement distributed tracing only but with Spring Boot-3 we use Micrometer, Open Telemetry & Zipkin to implement distributed tracing, logging and metrices.

To implement the Micrometer, Open Telemetry & Zipkin add the 3 dependencies and related properties as in currency-exchange service.





To start the Zipkin server use below command

docker run -d -p 9411:9411 openzipkin/zipkin:2.23