Spring Boot Actuator provides dependency management and auto-configuration for Micrometer, an application metrics facade that supports numerous monitoring systems like Prometheus.

Micrometer is like the slf4j in log world.

Spring boot actuator document:

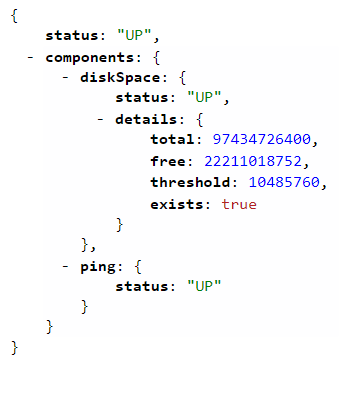
<https://docs.spring.io/spring-boot/docs/current/reference/html/production-ready-features.html#production-ready-enabling>

1. Create a spring boot project in <https://start.spring.io/.>

<dependency>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-actuator</artifactId>  
 </dependency>

1. Test the health endpoint.(the jsonview chrome plugin is suggested)

<http://localhost:8080/actuator/health>



1. Integrate prometheus

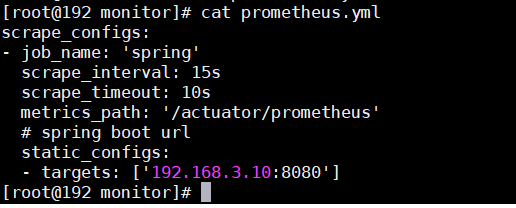
Add micrometer-registry-prometheus dependency

<dependency>  
 <groupId>io.micrometer</groupId>  
 <artifactId>micrometer-registry-prometheus</artifactId>  
 </dependency>

1. Test the prometheus endpoint

<http://localhost:8080/actuator/prometheus>

1. Run the prometheus



docker run \

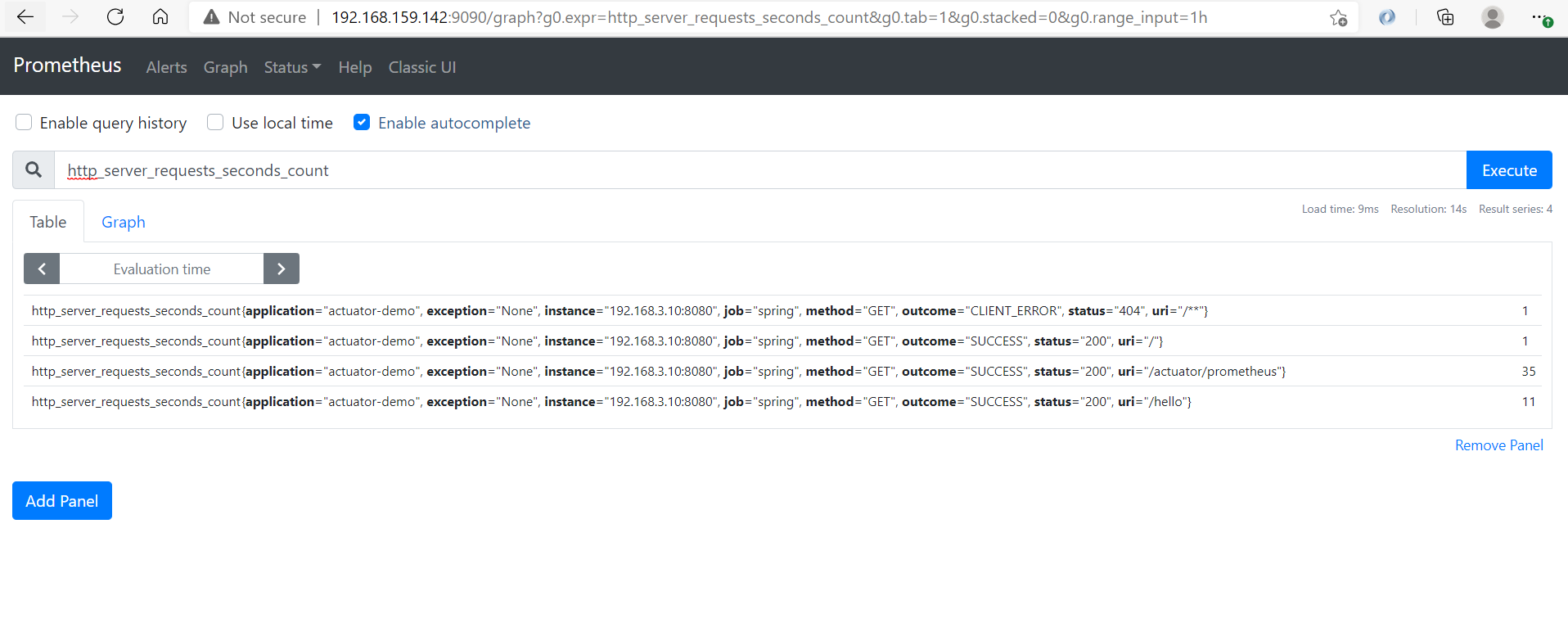
-p 9090:9090 \

-v /root/software/monitor/prometheus.yml:/etc/prometheus/prometheus.yml \

prom/prometheus

[http://192.168.159.142:9090/](http://192.168.159.142/) is prometheus url.

<http://192.168.3.10:8080/> is spring boot url.



1. Install grafana(to have a better dashboard ui experience)

docker run -d -p 3000:3000 grafana/grafana

<http://192.168.159.142:3000/>

Add Data Source

Add dashboard

Note:

1. <http://localhost:8080/actuator/prometheus>

Exposes metrics in a format that can be scraped by a Prometheus server. Requires a dependency on micrometer-registry-prometheus.

1. Prometheus actuator endpoint

This endpoint is enabled when the micrometer-registry-prometheus jar is added.

1. Status heathy

The status will be UP as long as the application is healthy. It will show DOWN if the application gets unhealthy due to any issue like connectivity with the database or lack of disk space etc.

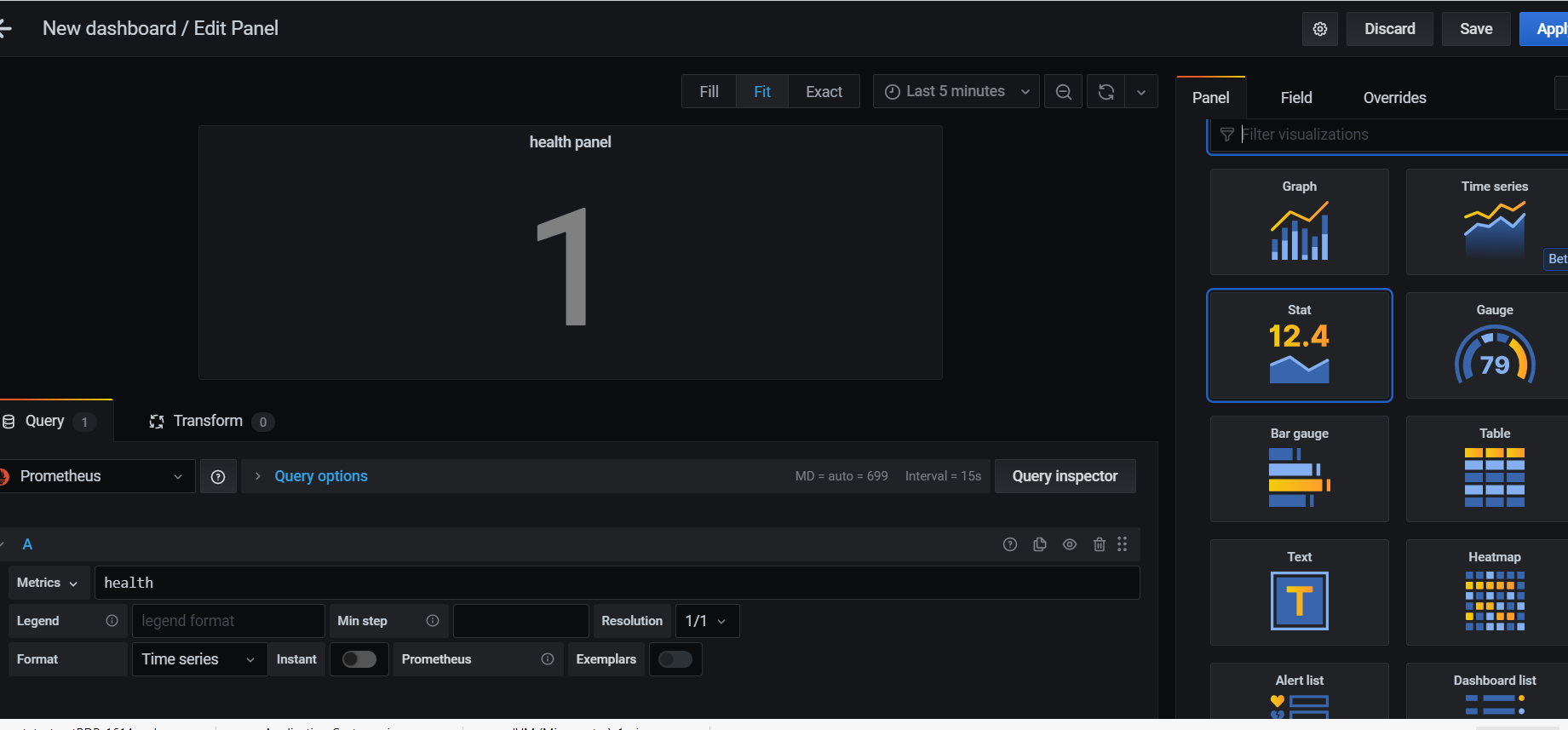
Grafana Status Panel

This is a plugin meant to be used as a centralized view for the status of component in a glance. It is very similar to the Single Stat panel, but it can hold multiple values from the same data source. Each value can be used to customize the panel in different ways:

<https://community.grafana.com/t/monitor-website-is-up-or-down-using-grafana/23251>

Suggest to use single stat panel.

Single Stat panel is built-in panel.



But it can not check when the health metrics for one service is empty, then set the status ‘down’.

And the spring boot actuator health indicator is not used to monitor the jvm down or up, it’s used to monitor the third party service is up or down, like the datasource.

docker run -id --name blackbox-exporter -p 9115:9115 prom/blackbox-exporter

http://192.168.159.147:9115/probe?target=192.168.3.10:8081&module=http\_2xx