Lab exercise: Prometheus showcase

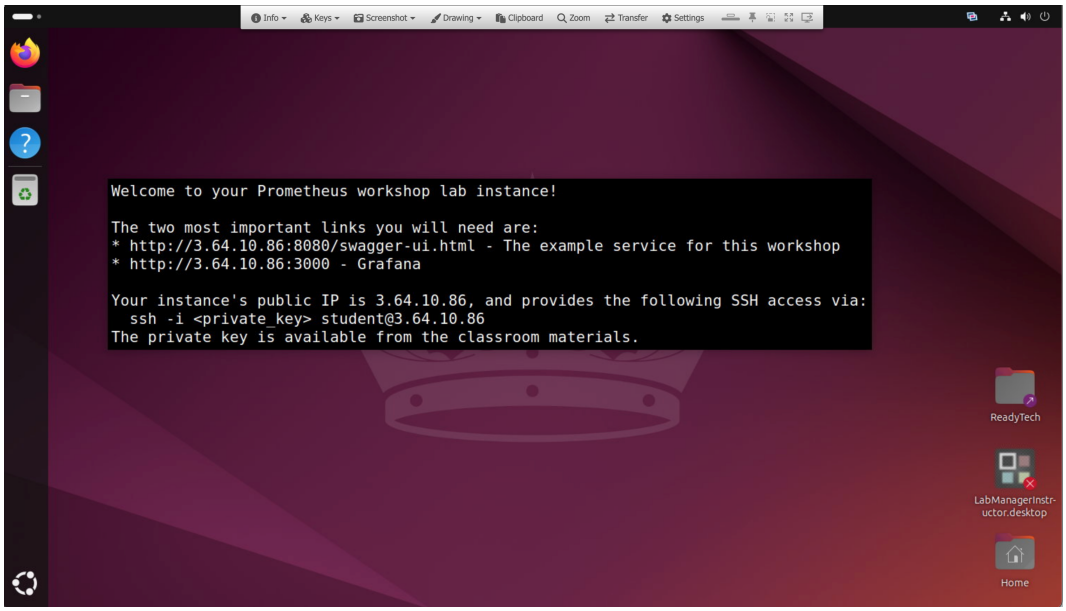
Overview

This lab is intended to familiarize you with your lab setup as well as the basics of using Grafana. As part of the workshop you are provided with your own self-contained lab environment; in this exercise you'll learn how to connect to your lab, interact with the sample Java service driving this workshop, and have your first taste of what Grafana is and can do.

Assignments

1. Setup your lab

Your initial connection to the lab environment happens via the classroom application, which provides an in-browser remote desktop experience (the username/password for logging into the lab environment is `student/student`). The desktop will show you a welcome message:



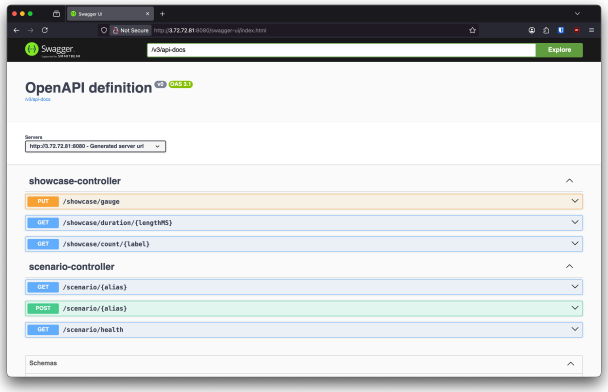
Although the lab is and will remain available for direct access through the classroom app, it can sometimes be easier to interact with the lab components from the comfort of your own laptop or desktop computer. The IP address for your lab is displayed on the Ubuntu desktop as shown in the above screenshot.

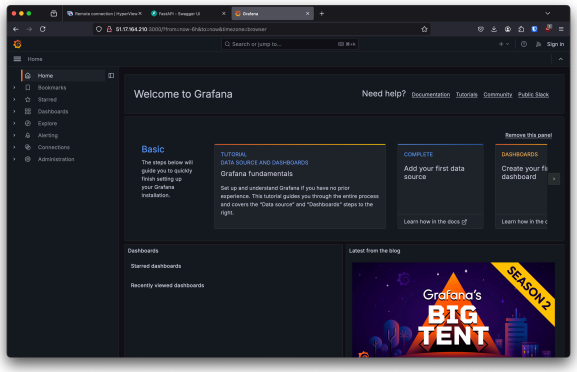
With this IP address you can connect directly to the various lab components:

- The sample service is available at `http://<lab_ip>:8080/swagger-ui.html`
- Grafana is available at `http://<lab_ip>:3000`
- Although you're not expected to need it, SSH access to the server is provided (the private key is available in the classroom materials from the classroom app)

You can also use the Firefox browser on your lab computer via the lab view, where (for convenience) you can omit the external IP and simply use `localhost`.

Please make sure you can access all the above components; the results should look similar to the following screenshots:





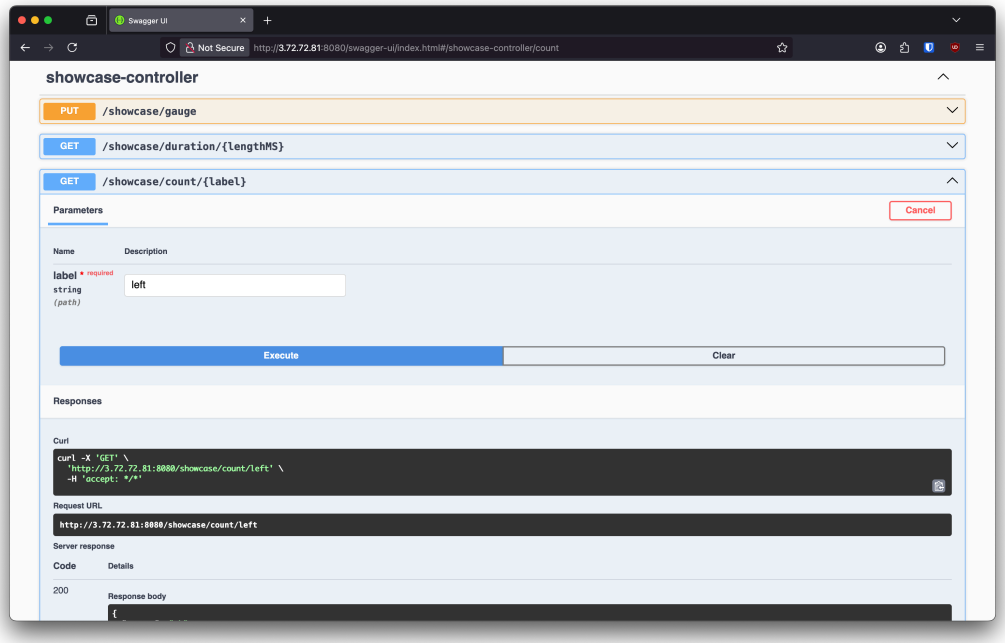
If you run into any trouble, please call on your instructor in the classroom for help!

2. Metric showcase

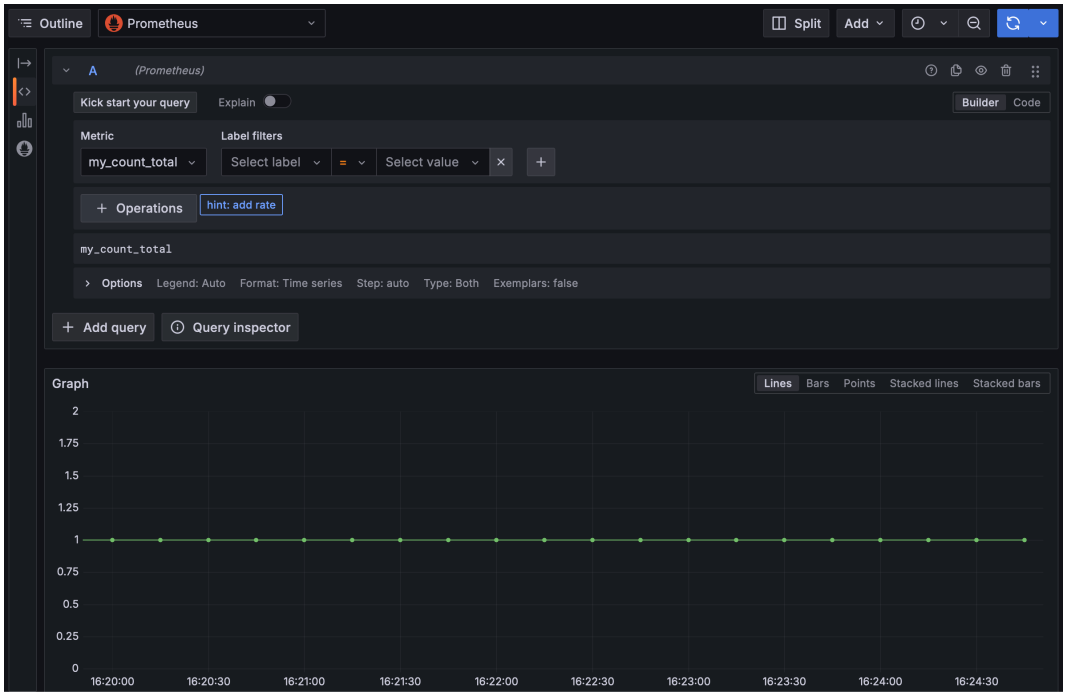
The Java-based sample service contains a set of endpoints to showcase the different metric types. You can use them directly from the Swagger documentation tab you've opened in the previous section; just don't forget to click on "Try it out" to enable the test UI.

2.1. Counters

To start you off, let's see an actual counter in action. Expand the `/showcase/count/{label}` endpoint; as you might expect, this endpoint simply increases a counter (with the associated label). Try running it a few times with different labels, just to get an initial data set into the system:



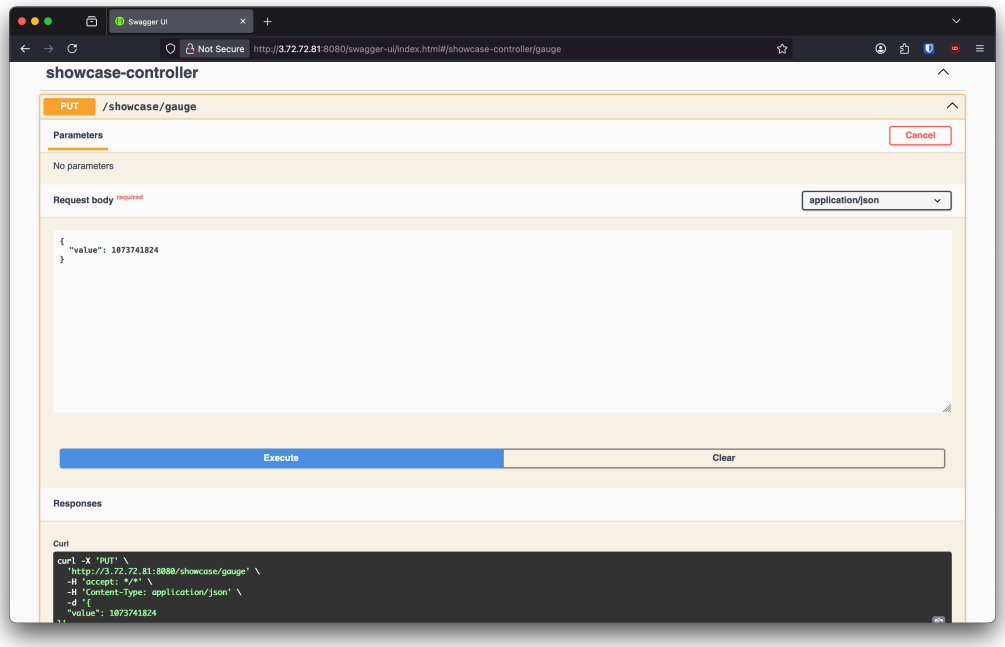
Now switch to Grafana (which, as a reminder, is available at `http://<lab_ip>:3000`). A great way to explore the available data set is using the Explore tab on the left and using the Metric field to get a feel for what's out there. You can also pop out the Metric Explorer which provides a nice listing of available time series with appropriate descriptions, but to start with let's select the showcase metric called `my_count_total` and click on "execute query":



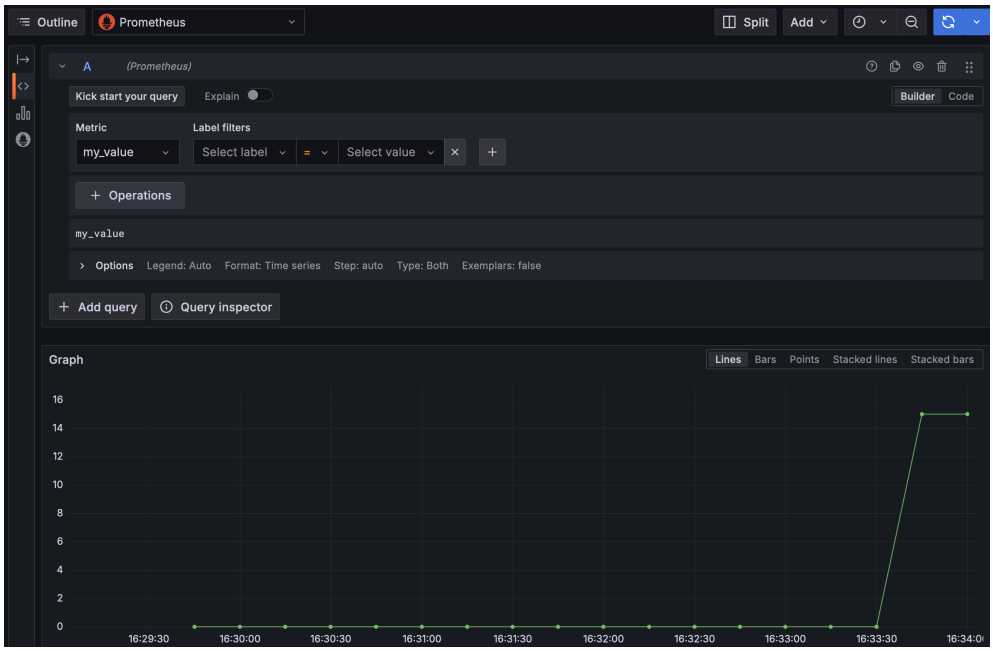
Now, run a few more requests with different labels and observe them being added to the chart! (Please allow up to a minute for new data to show up).

2.2. Gauges

Similarly to the previous example, the `PUT /showcase/gauge` endpoint lets you set the `my_value` metric to an arbitrary numeric value. Since there's no initial value, this metric won't show up on Grafana at first – you need to initialize it first by executing the endpoint with some value (0 by default), which you can do through the Swagger web interface:

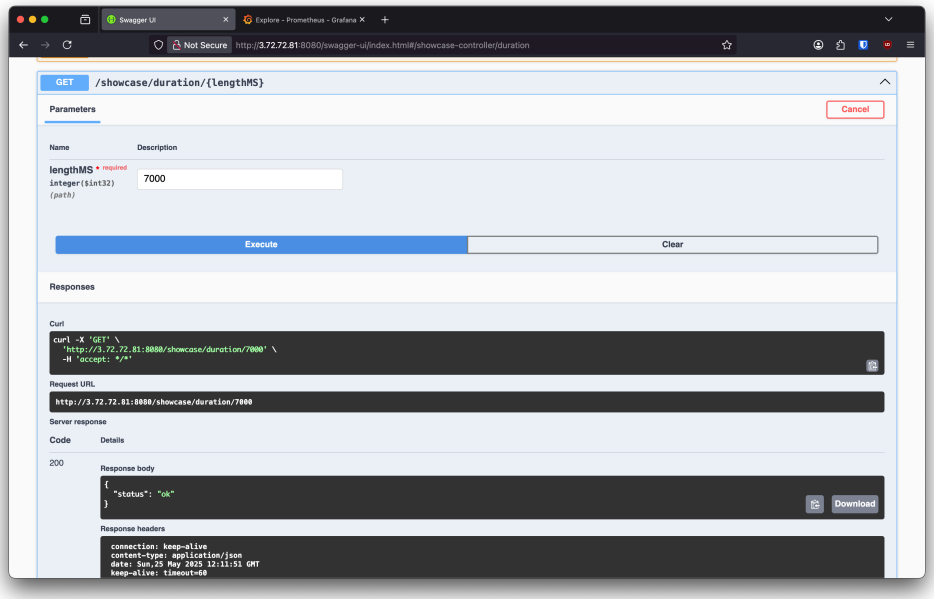


Now, wait a bit (say half a minute) and set the value to something else. Repeat a couple of times and switch back to Grafana to chart the `my_value` metric to see your changes, which may take a few seconds to reflect:

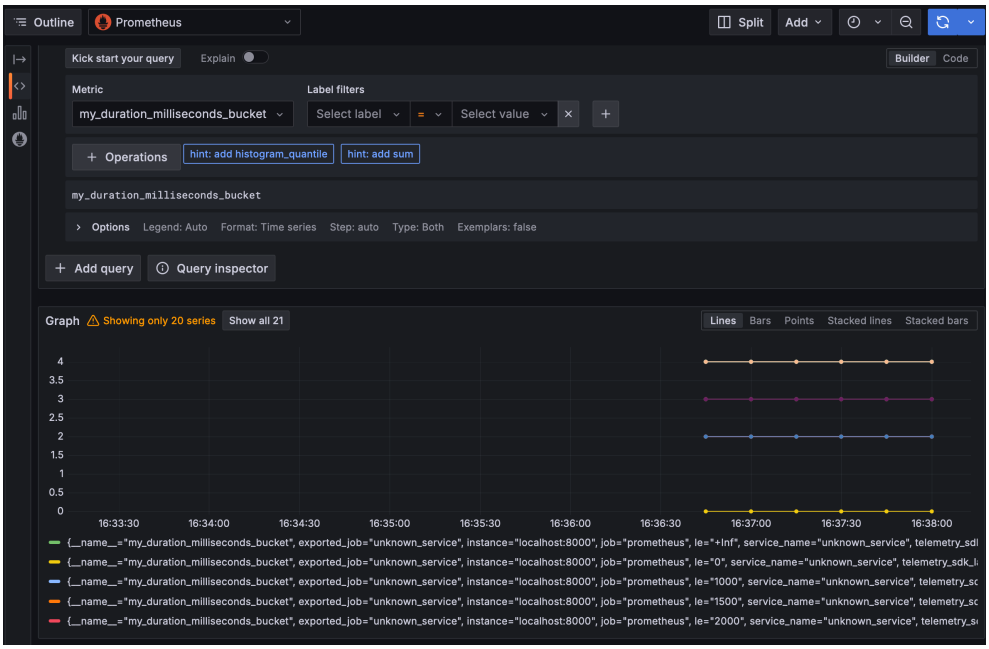


2.3. Histograms

Finally, the /showcase/duration/{length_ms} endpoint records an arbitrary number of durations (the length of which is specified by the parameter) as a *distribution*. Try running a few numbers through the endpoint:



Next, switch back to Grafana and chart the my_duration_milliseconds_bucket metric. You'll note that this results in *several* time series, with the `le` label (short for "lesser or equal to") providing the distribution. For example, with four durations of 70, 250, 2500 and 7000ms:



3. Stretch goals

Use Grafana's Explore mode to figure out suitable metrics that could answer the following questions (you don't have to come up with the formulae, that's for the next lab! Just find the metrics themselves):

- System disk space utilization
- The size of HTTP server responses
- How many HTTP requests are actively served
- System CPU utilization

References

- Workshop presentation in class materials
- SSH private key for connecting to your lab is available in the class materials
- The full sources for everything can always be found here: https://github.com/holograph/prometheus-workshop-service-java