**Table of Contents**

Exercise – 1: Collecting Metrics: 2

Visualising Data 5

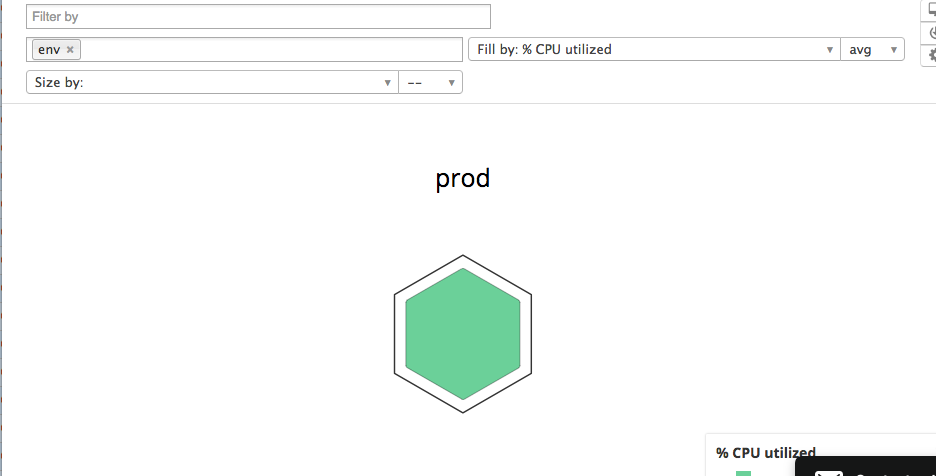
Monitoring Data 7

Collecting APM Data 10

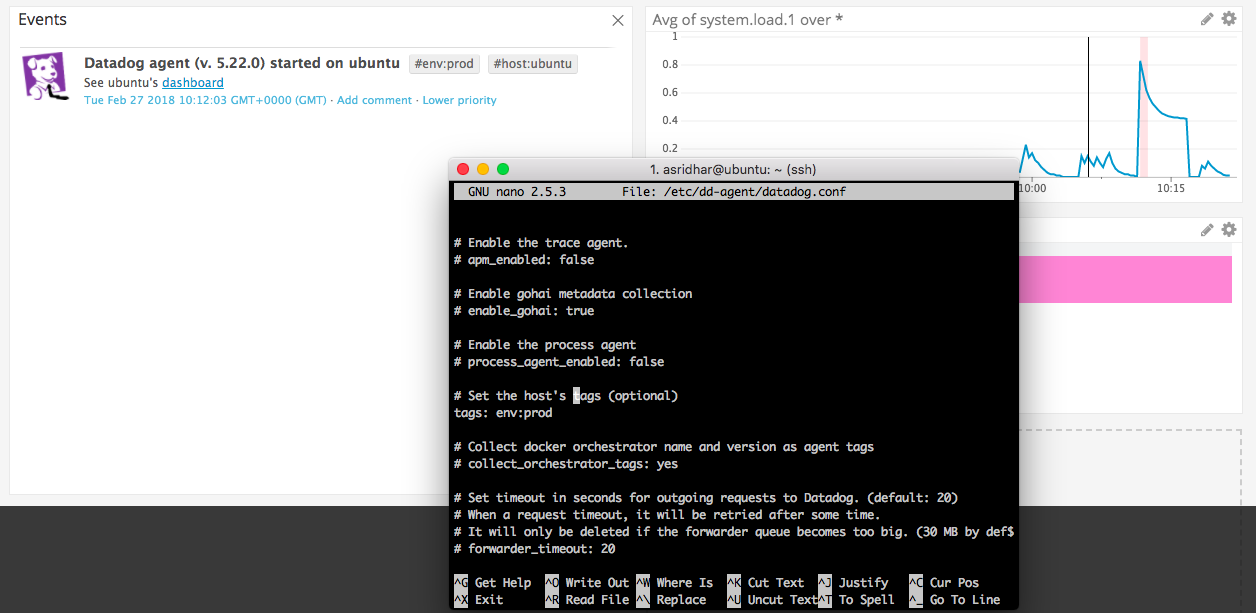
# Exercise – 1: Collecting Metrics:

* Add tags in the Agent config file and show us a screenshot of your host and its tags on the Host Map page in Datadog.

**Image: Host Map page in Datadog**

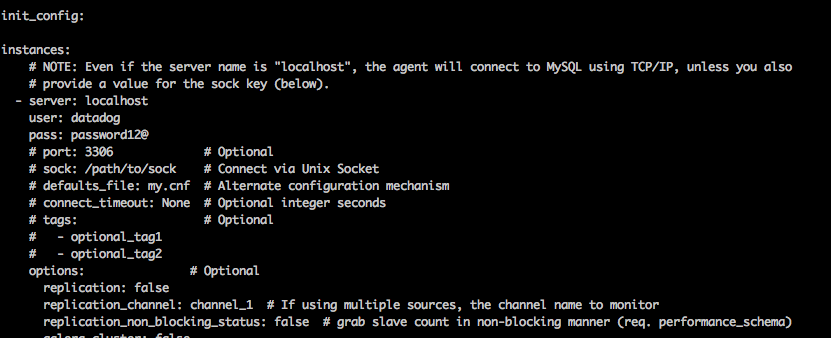


**Image: Agent Configuration**

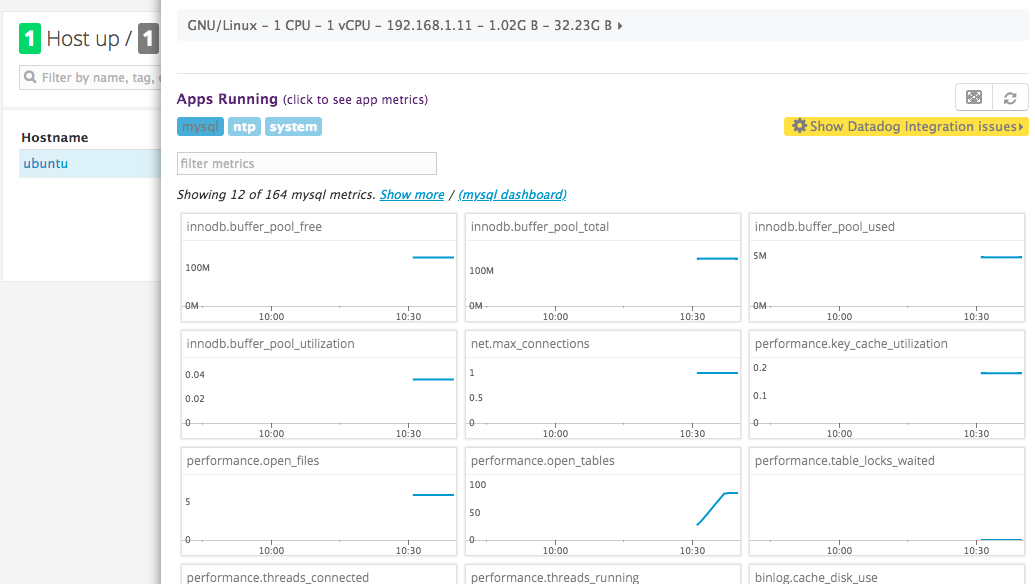


* Install a database on your machine (MongoDB, MySQL, or PostgreSQL) and then install the respective Datadog integration for that database.

**Image: MySQL Agent configuration**

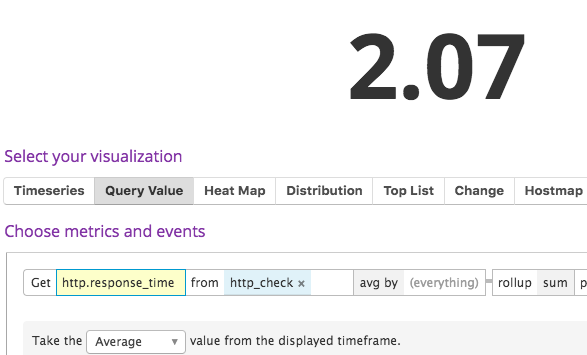
****

**Image: MySQL Datadog**

****

* Create a custom Agent check that submits a metric named my\_metric with a random value between 0 and 1000.

Created a http agent check, from documentation which submits the http response time as a random value between 0 and 1000

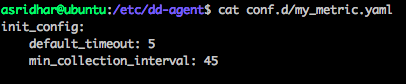
**Image: http response being recorded**

* Change your check's collection interval so that it only submits the metric once every 45 seconds.

Updated the my\_metric.yaml file to set min\_collection\_interval value to 45

* **Bonus Question** Can you change the collection interval without modifying the Python check file you created?

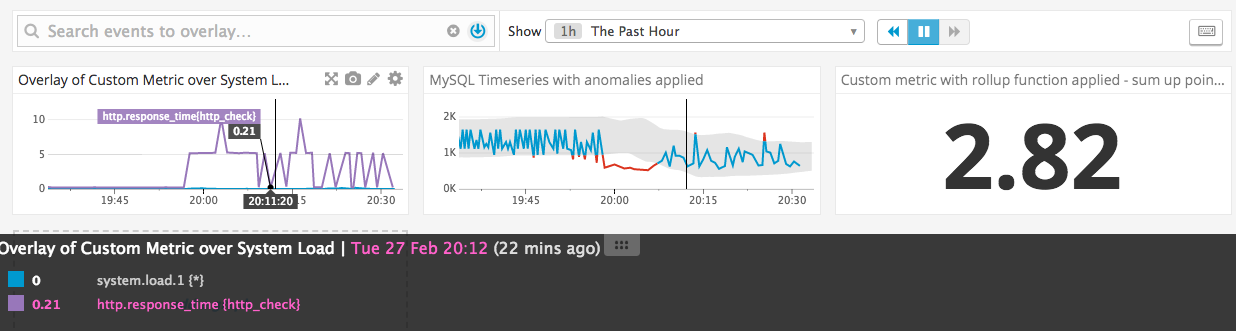
Yes, see screenshot below:

**Image YAML File**

# Visualising Data

* Utilize the Datadog API to create a Timeboard that contains:
* Your custom metric scoped over your host.
* Any metric from the Integration on your Database with the anomaly function applied.
* Your custom metric with the rollup function applied to sum up all the points for the past hour into one bucket

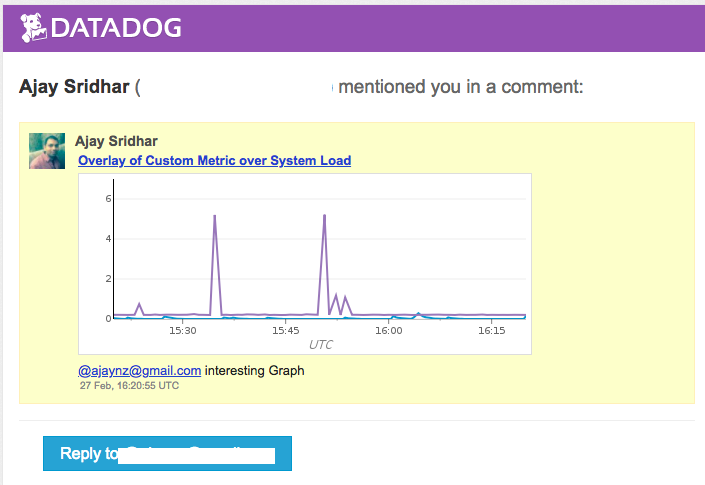
**Image: Timeboard covering all points above**



* Please be sure, when submitting your hiring challenge, to include the script that you've used to create this Timemboard.

Please see createdash.sh

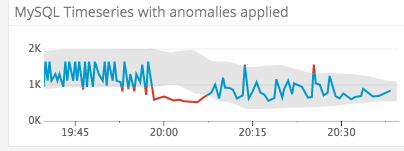
* Once this is created, access the Dashboard from your Dashboard List in the UI:
* Set the Timeboard's timeframe to the past 5 minutes
  + Its not possible to set timeframe to last 5 minuets, the filter only go as far back as 1hour
* Take a snapshot of this graph and use the @ notation to send it to yourself.

**Image: Mentions via Email**

* **Bonus Question**: What is the Anomaly graph displaying?

The anaomaly graph tracks deviations – as per example below, the red line denotes the anomalies tracked by datadog from the data points gathered by the MySQL agent:

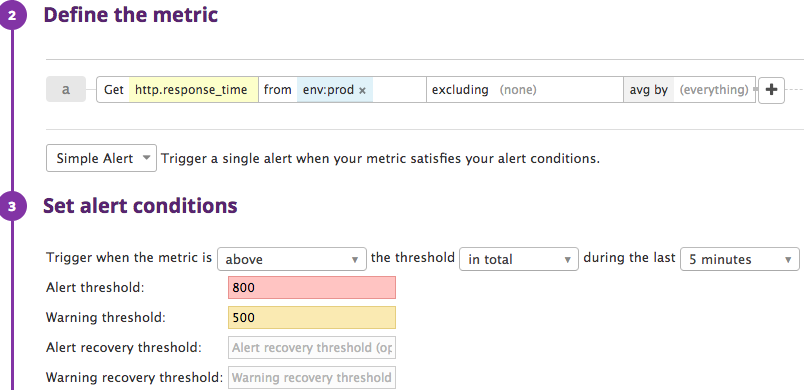
**Image: Graph of anomalies**



# Monitoring Data

* Create a new Metric Monitor that watches the average of your custom metric (my\_metric) and will alert if it’s above the following values over the past 5 minutes:
* Warning threshold of 500
* Alerting threshold of 800
* And also ensure that it will notify you if there is No Data for this query over the past 10m.

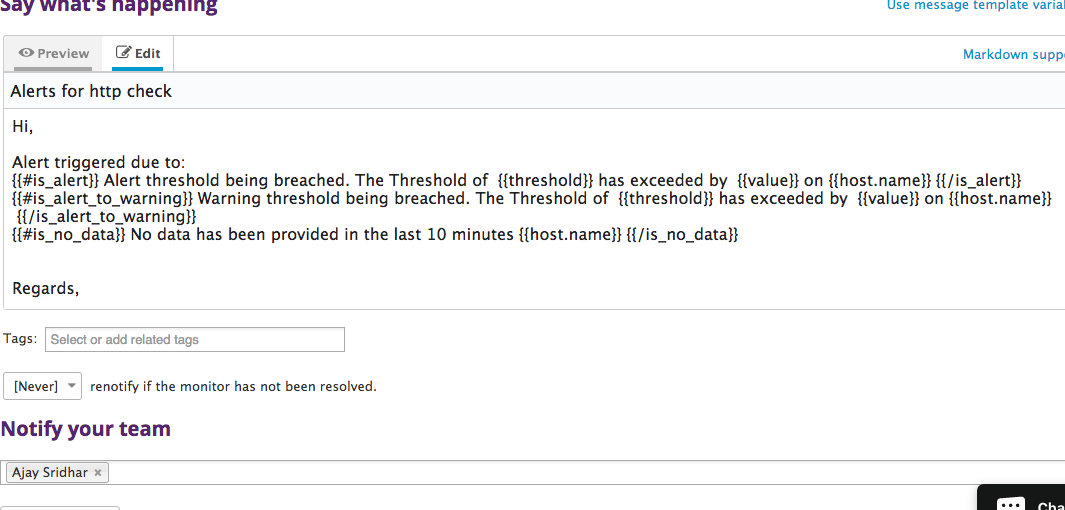
**Image: Metric Monitor with above parameters**



* Please configure the monitor’s message so that it will:
* Send you an email whenever the monitor triggers.
* Create different messages based on whether the monitor is in an Alert, Warning, or No Data state.
* Include the metric value that caused the monitor to trigger and host ip when the Monitor triggers an Alert state.

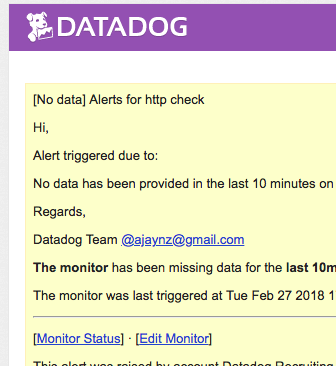
**\*\* Please note, there doesn’t seem to be anyway to access Host IP address via variables (but only host names), so this data isn’t recorded \*\***

**Image: configuration for notification**

****

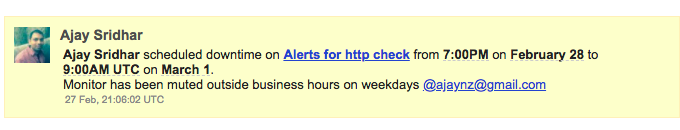
* When this monitor sends you an email notification, take a screenshot of the email that it sends you.

**Image: Email notification on lack of data stream for over 10 minutes**



* **Bonus Question**: Since this monitor is going to alert pretty often, you don’t want to be alerted when you are out of the office. Set up two scheduled downtimes for this monitor:
* One that silences it from 7pm to 9am daily on M-F,
* And one that silences it all day on Sat-Sun.
* Make sure that your email is notified when you schedule the downtime and take a screenshot of that notification.

**Image: Downtime schedule**



# Collecting APM Data

* Provide a link and a screenshot of a Dashboard with both APM and Infrastructure Metrics.

The provided dashboard below has the following aim, it lists the total customer visits and website reponse times over the last hour.

The dashboard also maps infrastructure/system utilisation against the application, such that its easy to debug slow response times against database, cpu and memory utilisation.

Please follow link to dashboard here;

<https://p.datadoghq.com/sb/efdb85d86-7883db2ceb38ccbad6b551857ccd8b9b>

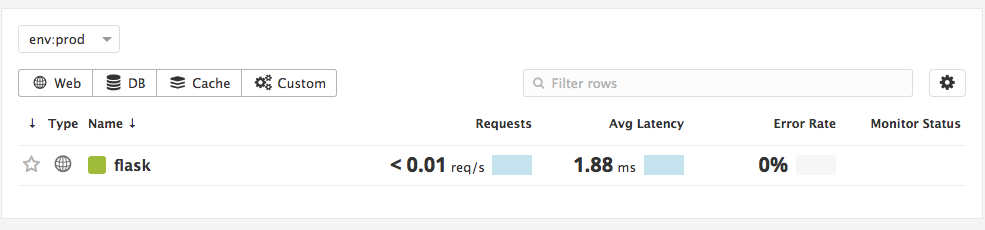
**\*\*\*Please note, the dashboard only goes back 60 minutes – depending on when the page is accessed there might not be data displaying as the agent might be offline. I have provided Dashboard Public.PNG file as supporting evidence to this point\*\***

* Please include your fully instrumented app in your submission, as well.

I used the provided flask application and the ddtrace-run python agent.

Provided **datadog.py file**

**Image: Flask traces**



* **Bonus Question**: What is the difference between a Service and a Resource?

A service is a grouping or named set of resources (Processes) which form a feature set. For inscance, a multi-threaded webapp or database can own multiple threads and belog to one service.

A resource is a a particular query or criteria attached to the service, for instance it could be a canonical URL: /employee/ or /API/