Observing Cloud Resources

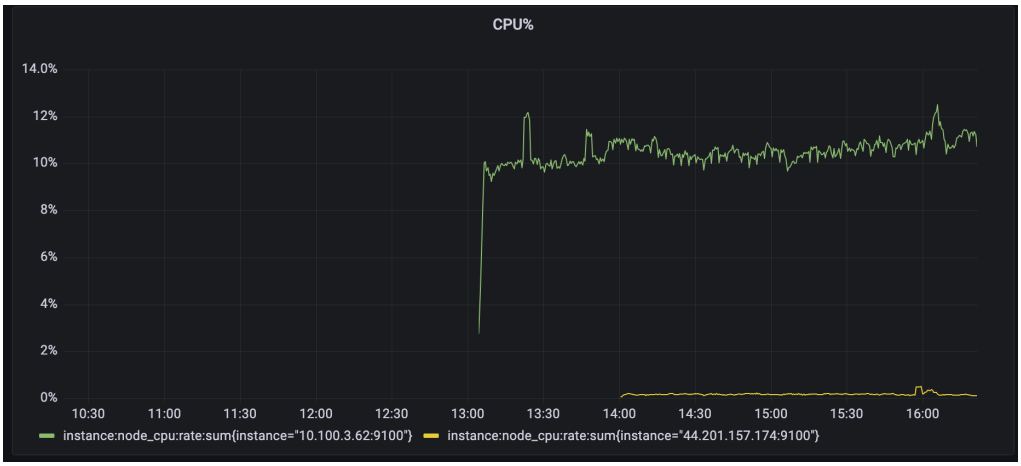
SRE Project Template

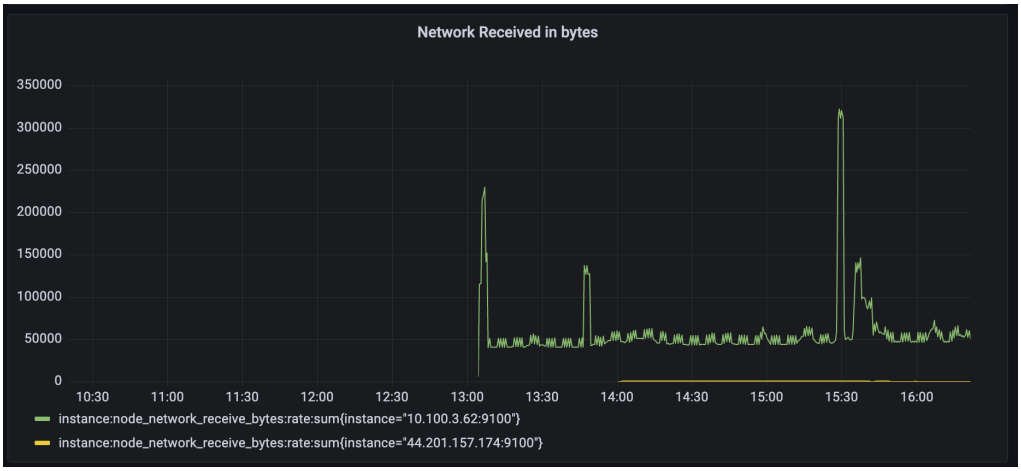
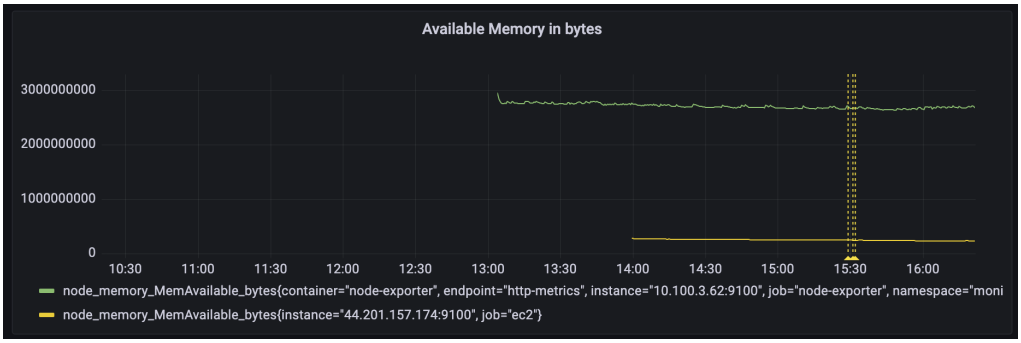
Categorize Responsibilities

Prometheus and Grafana Screenshots
Provide a screenshot of the Prometheus node_exporter service running on the EC2 instance. Use the following command to show that the system is running: `sudo systemctl status node_exporter`

```
ubuntu@ip-172-31-56-216:~$ sudo systemctl status node_exporter
● node_exporter.service - Node Exporter
     Loaded: loaded (/etc/systemd/system/node_exporter.service; enabled; vendor preset: enabled)
     Active: active (running) since Mon 2022-04-25 11:58:58 UTC; 4s ago
   Main PID: 30364 (node_exporter)
      Tasks: 4 (limit: 1109)
     CGroup: /system.slice/node_exporter.service
             └─30364 /usr/local/bin/node_exporter

Apr 25 11:58:58 ip-172-31-56-216 node_exporter[30364]: level=info ts=2022-04-25T11:58:58.031Z caller=node_exporter.go:115 collector=the
Apr 25 11:58:58 ip-172-31-56-216 node_exporter[30364]: level=info ts=2022-04-25T11:58:58.031Z caller=node_exporter.go:115 collector=tim
Apr 25 11:58:58 ip-172-31-56-216 node_exporter[30364]: level=info ts=2022-04-25T11:58:58.031Z caller=node_exporter.go:115 collector=tim
Apr 25 11:58:58 ip-172-31-56-216 node_exporter[30364]: level=info ts=2022-04-25T11:58:58.031Z caller=node_exporter.go:115 collector=udp
Apr 25 11:58:58 ip-172-31-56-216 node_exporter[30364]: level=info ts=2022-04-25T11:58:58.031Z caller=node_exporter.go:115 collector=una
Apr 25 11:58:58 ip-172-31-56-216 node_exporter[30364]: level=info ts=2022-04-25T11:58:58.031Z caller=node_exporter.go:115 collector=vms
Apr 25 11:58:58 ip-172-31-56-216 node_exporter[30364]: level=info ts=2022-04-25T11:58:58.031Z caller=node_exporter.go:115 collector=xfs
Apr 25 11:58:58 ip-172-31-56-216 node_exporter[30364]: level=info ts=2022-04-25T11:58:58.031Z caller=node_exporter.go:115 collector=zfs
Apr 25 11:58:58 ip-172-31-56-216 node_exporter[30364]: level=info ts=2022-04-25T11:58:58.031Z caller=node_exporter.go:199 msg="Listenin
Apr 25 11:58:58 ip-172-31-56-216 node_exporter[30364]: level=info ts=2022-04-25T11:58:58.033Z caller=tls_config.go:191 msg="TLS is disa
ubuntu@ip-172-31-56-216:~$
```

Host Metric (CPU, RAM, Disk, Network)	Dashboard
CPU	

Network	
Disk I/O	
Memory	

Responsibilities

1. The development team wants to release an emergency hotfix to production. Identify two roles of the SRE team who would be involved in this and why.

An **Infrastructure engineer** because he does 50% development tasks and 50 % operations tasks. His expertise is required to gather knowledge of the hotfix and its impact on the release.
A **Release manager** because his roles execute the release, and rollback procedures.

2. The development team is in the early stages of planning to build a new product. Identify two roles of the SRE team that should be invited to the meeting and why.

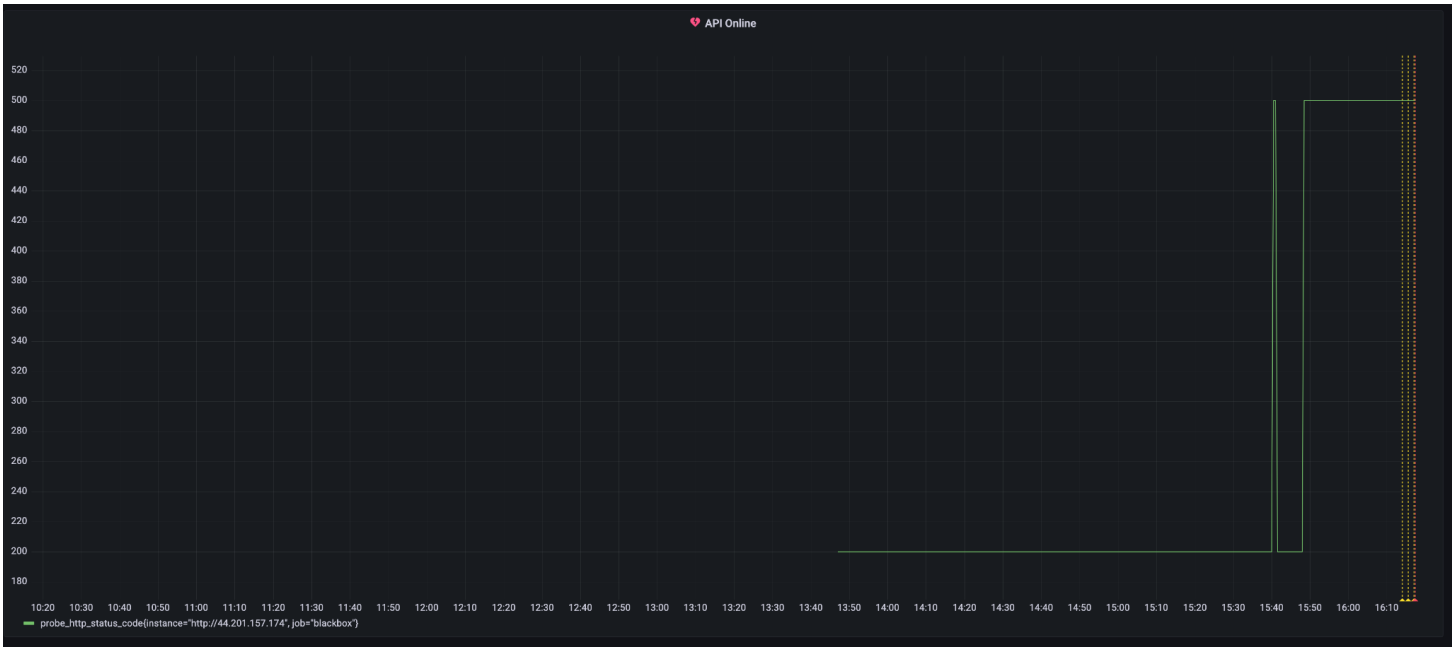
A **Team lead** to help the development team by directing the work, forming workflows and coordinating the development around the objective.

A **System Architect** to make recommendations about the impact and requirements of the new product, but also provide if necessary additional infrastructure components to the development team.
3. The emergency hotfix from question 1 was applied and is causing major issues in production. Which SRE role would primarily be involved in mitigating these issues?
A **Release manager** would primarily be involved.

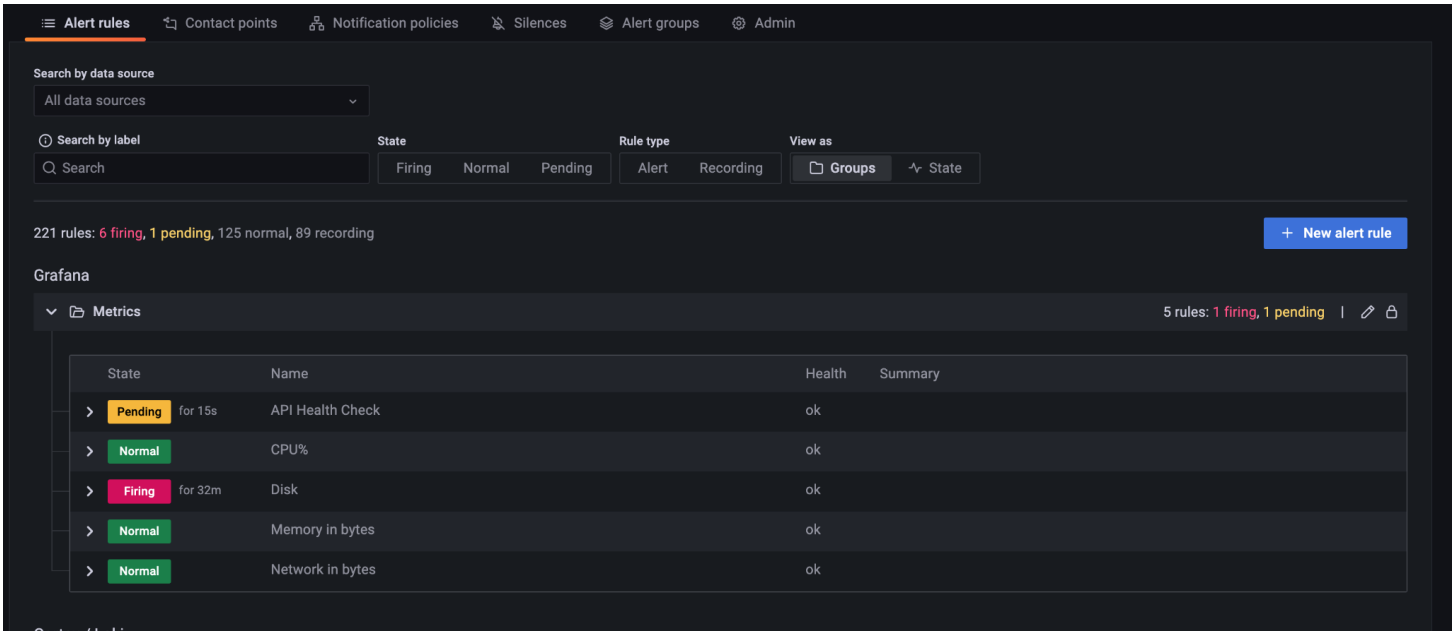
Team Formation and Workflow Identification

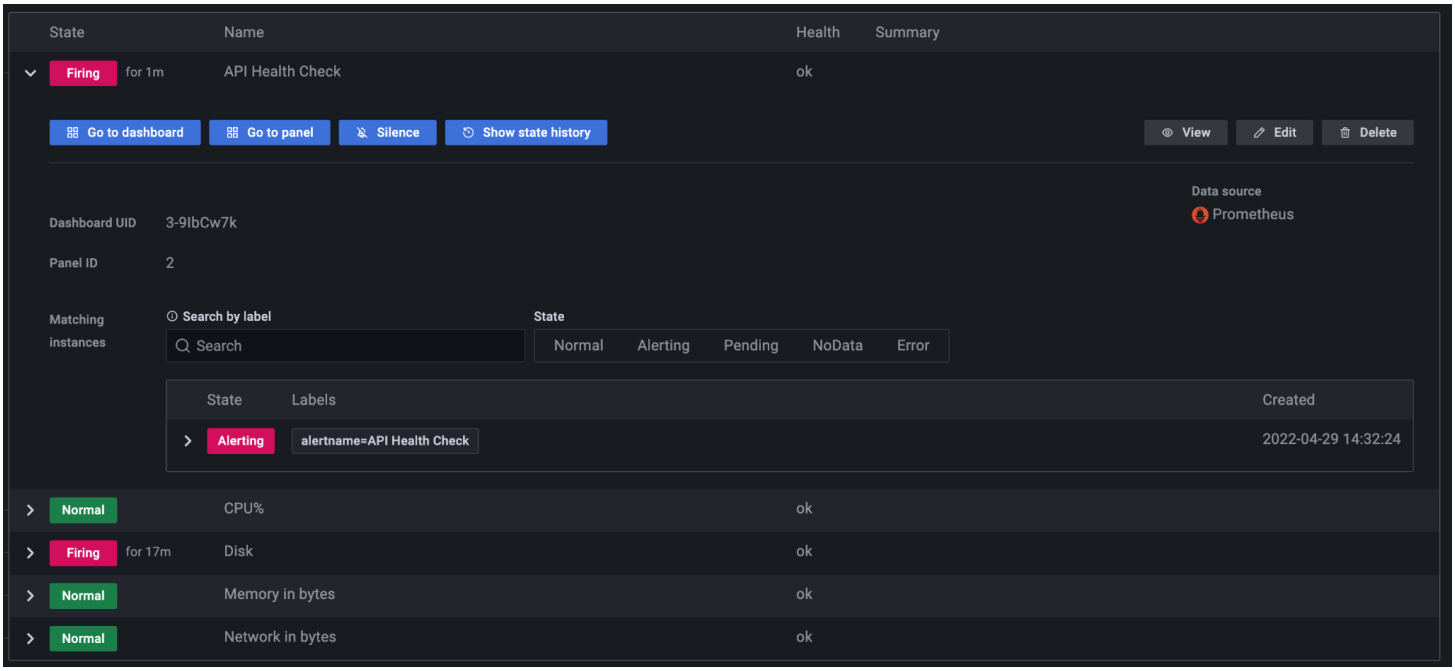
API Monitoring and Notifications

Display the status of an API endpoint: Provide a screenshot of the Grafana dashboard that will show at which point the API is unhealthy (non-200 HTTP code), and when it becomes healthy again (200 HTTP code).



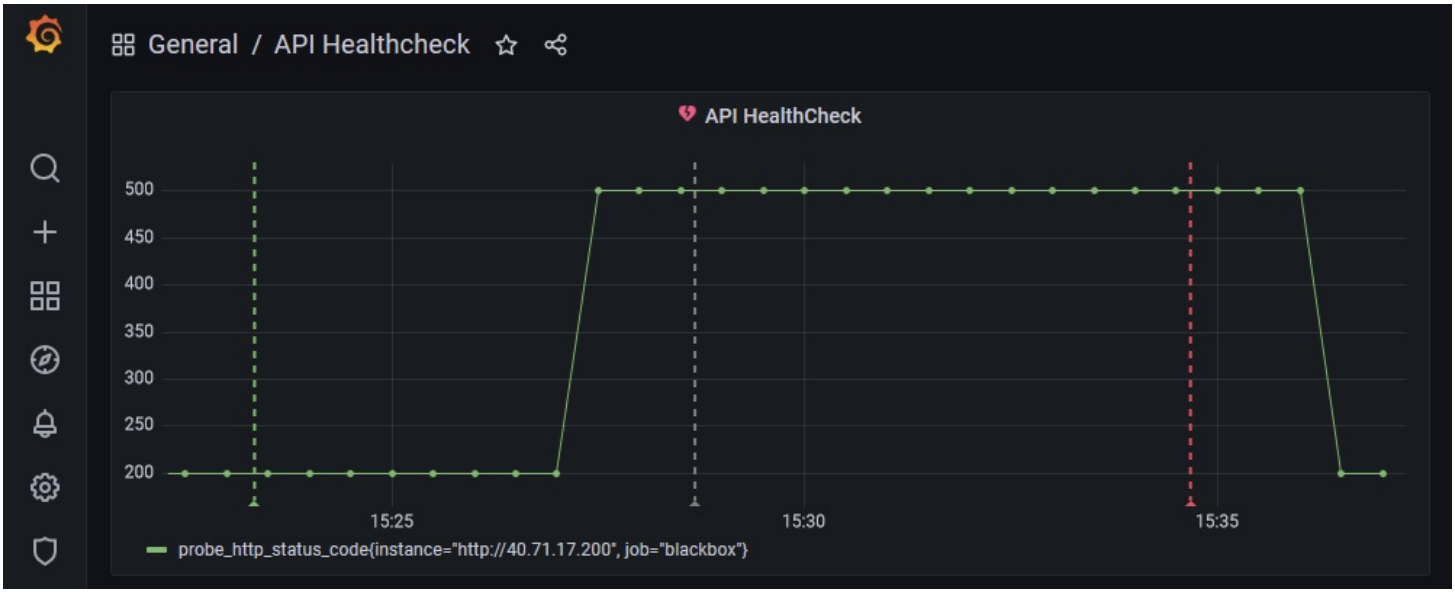
Create a notification channel: Provide a screenshot of the Grafana notification which shows the summary of the issue and when it occurred.



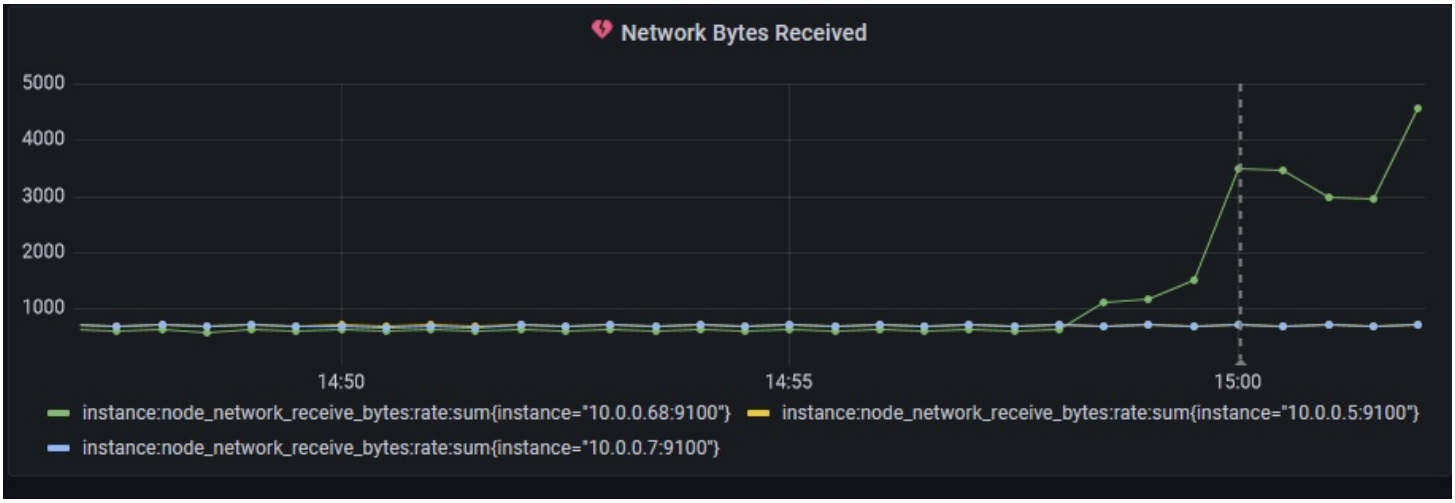


Configure alert rules: Provide a screenshot of the alert rules list in Grafana.

Applying the Concepts

Graph 1

4a. Given the above graph, where does it show that the API endpoint is down? Where on the graph does this show that the API is healthy again?
It shows the API endpoint was down at 15:27:30" when the status code changes from 200 to 500. Then it stays down till 15:36, and we get an healthy signal at 15:36:30"
4b. If there was no SRE team, how would this outage affect customers?
It depends if the instance is the only resource which has this API, if there's a load balancer upfront the API, the client which hits the API will use the other resource attached to the Load Balancer. If the instance is the only resource which has the API, then the customer will suffer an outage related to this failure.
4c. What could be put in place so that the SRE team could know of the outage before the customer does?
Synthetic monitoring with alerts all the time to identify very quickly and inform the corresponding team.

Graph 2



5a. Given the above graph, which instance had the increase in traffic, and approximately how many bytes did it receive (feel free to round)?

*The **10.0.0.68** instance had an increase in traffic. At the peak it shows an average of **4700** bytes received.*

5b. Which team members on the SRE team would be interested in this graph and why?

A **System Architect** would be interested. He would know if the current infrastructure is strong enough to handle the traffic.
A **Monitoring Engineer** would be interested in this graph. With the proper alerts configured He would be able to communicate in an efficient way about latency.
A **Release Manager** would be interested as well to know the impacts of the releases of the infrastructure.