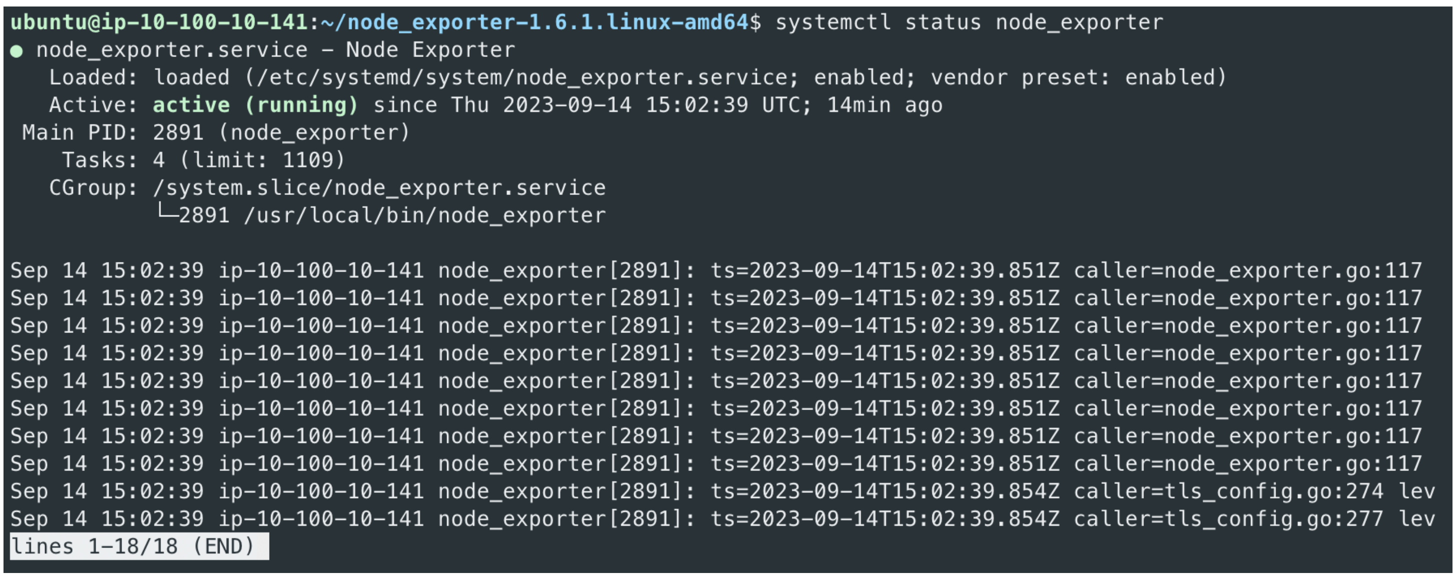
Observing Cloud Resources

SRE Project Template

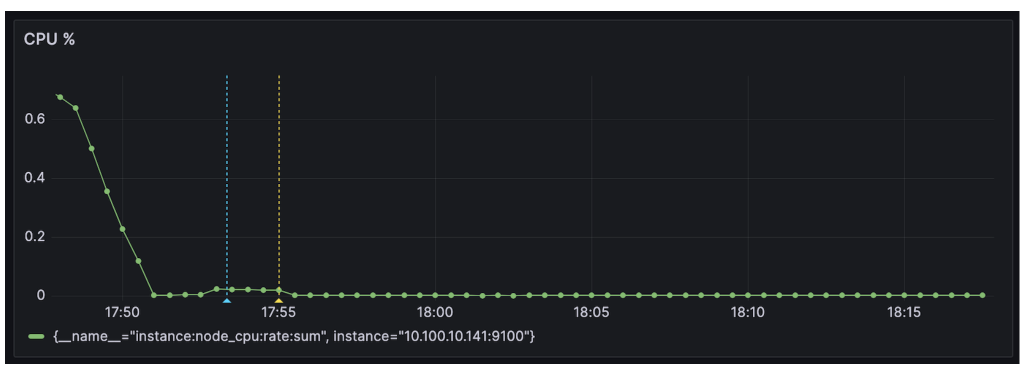
Categorize Responsibilities

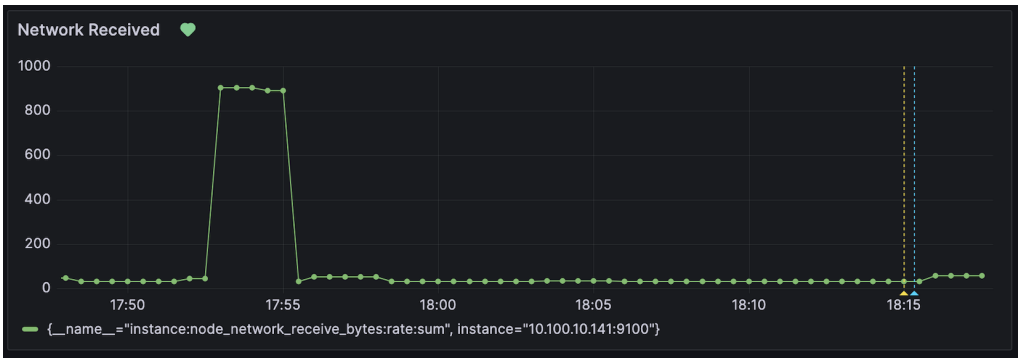
Prometheus and Grafana Screenshots
Provide a screenshot of the Prometheus node_exporter service running on the EC2 instance. Use the following command to show that the system is running: `sudo systemctl status node_exporter`



```
ubuntu@ip-10-100-10-141:~/node_exporter-1.6.1.linux-amd64$ 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 Thu 2023-09-14 15:02:39 UTC; 14min ago
 Main PID: 2891 (node_exporter)
    Tasks: 4 (limit: 1109)
   CGroup: /system.slice/node_exporter.service
           └─2891 /usr/local/bin/node_exporter

Sep 14 15:02:39 ip-10-100-10-141 node_exporter[2891]: ts=2023-09-14T15:02:39.851Z caller=node_exporter.go:117
Sep 14 15:02:39 ip-10-100-10-141 node_exporter[2891]: ts=2023-09-14T15:02:39.851Z caller=node_exporter.go:117
Sep 14 15:02:39 ip-10-100-10-141 node_exporter[2891]: ts=2023-09-14T15:02:39.851Z caller=node_exporter.go:117
Sep 14 15:02:39 ip-10-100-10-141 node_exporter[2891]: ts=2023-09-14T15:02:39.851Z caller=node_exporter.go:117
Sep 14 15:02:39 ip-10-100-10-141 node_exporter[2891]: ts=2023-09-14T15:02:39.851Z caller=node_exporter.go:117
Sep 14 15:02:39 ip-10-100-10-141 node_exporter[2891]: ts=2023-09-14T15:02:39.851Z caller=node_exporter.go:117
Sep 14 15:02:39 ip-10-100-10-141 node_exporter[2891]: ts=2023-09-14T15:02:39.851Z caller=node_exporter.go:117
Sep 14 15:02:39 ip-10-100-10-141 node_exporter[2891]: ts=2023-09-14T15:02:39.851Z caller=node_exporter.go:117
Sep 14 15:02:39 ip-10-100-10-141 node_exporter[2891]: ts=2023-09-14T15:02:39.854Z caller=tls_config.go:274 lev
Sep 14 15:02:39 ip-10-100-10-141 node_exporter[2891]: ts=2023-09-14T15:02:39.854Z caller=tls_config.go:277 lev
lines 1-18/18 (END)
```

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

RAM	
Disk I/O	
Network Received	

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.

Release Manager: This role is responsible for managing code releases, including emergency hotfixes. The Release Manager would ensure that the hotfix has all its dependencies satisfied and coordinate the process of deploying the hotfix to production. They would use CI/CD tools to manage the release and have rollback procedures ready in case something goes wrong.

Infrastructure Engineer: While the Release Manager takes care of the deployment, the Infrastructure Engineer ensures that the infrastructure can support the new hotfix. They can be involved in automating operations tasks needed for the hotfix to work as expected, such as modifying configurations or setting up new resources. They would also be responsible for executing system patches or updates that might be required to support the hotfix.

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.

System Architect: Involving the System Architect is important for ensuring that the infrastructure for the new product will be scalable, robust, and cost-effective. They would make recommendations on the types of technologies to use and provide a migration path for integrating the new product into existing infrastructure. *Team Lead: The Team Lead's participation would help to align the SRE team's efforts with the development team's objectives for the new product. They would contribute to the overall architecture and strategy discussions.*
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?
Monitoring Engineer: Usually the first to know of an incident, the Monitoring Engineer would likely be the one to catch the problems caused by the hotfix through monitoring dashboards and alerting rules. They would be responsible for diagnosing what is going wrong and might work closely with the Infrastructure Engineer and Release Manager to roll back the changes if necessary.

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.

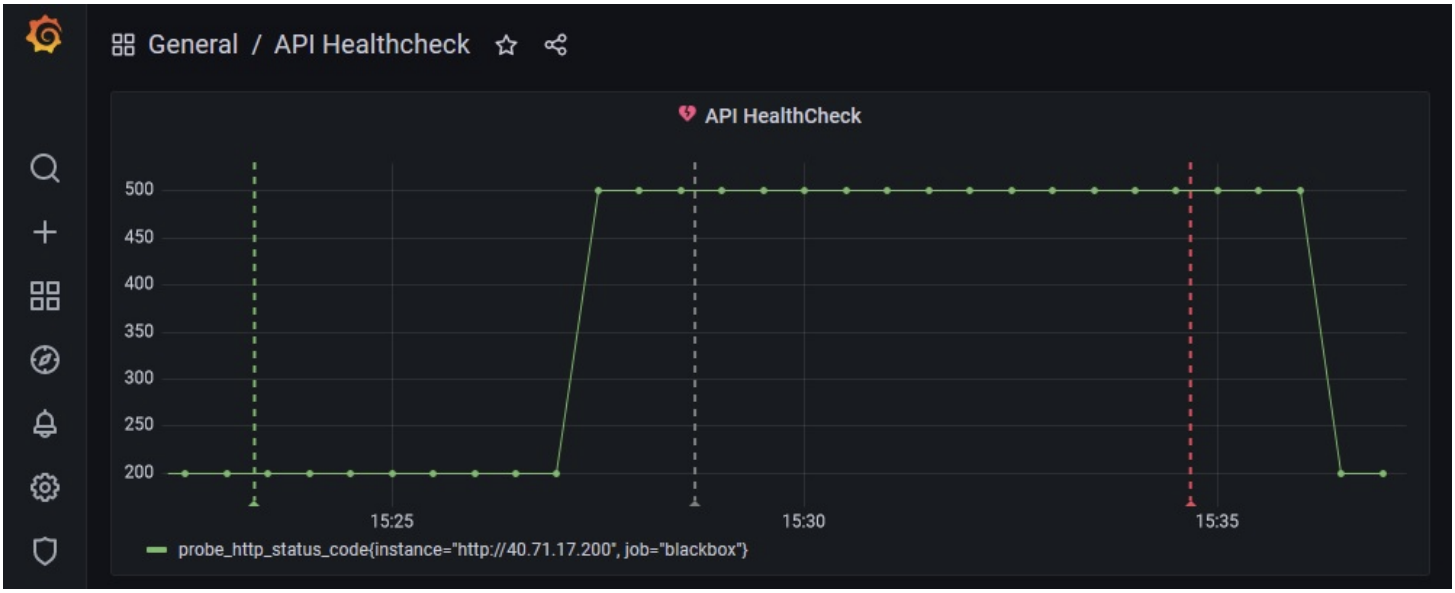
[FIRING:5] Flask App Latency Udacity - EC2 Instance (http://18.188.233.96 blackbox)
Firing

Value: A=0, C=1
Labels:
- alertname = Flask App Latency
- grafana_folder = Udacity - EC2 Instance
- instance = http://18.188.233.96
- job = blackbox
- phase = connect
Annotations:
- summary = The Flask App is facing high latency.
Source: http://localhost:3000/alerting/grafana/f63b430a-86e0-4c14-b64e-1d8fc9c31e87/view?orgId=1
Silence: http://localhost:3000/alerting/silence/new?alertmanager=grafana&matcher=alertname%3DFlask+App+Latency&matcher=grafana_folder%3DUdacity+-+EC2+Instance&matcher=instance%3Dhttp%3A%2F%2F18.188.233.96&matcher=job%3Dblackbox&matcher=phase%3Dconnect&orgId=1
Dashboard: http://localhost:3000/d/xtkCtBkiz?orgId=1
Panel: http://localhost:3000/d/xtkCtBkiz?orgId=1&viewPanel=25

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

I already tore down the resources, but let me know if it's absolutely required to re-deploy everything to take this screenshot.

Applying the Concepts

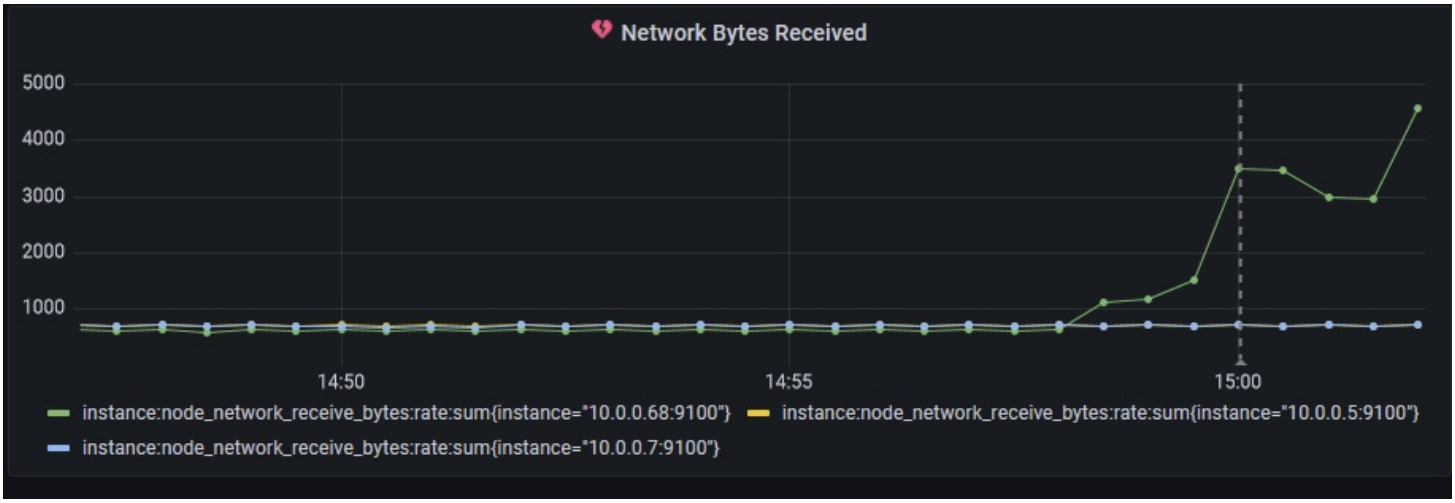
Graph 1

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?
The API was down around 15:27 and it came back up at about 15:36.
4b. If there was no SRE team, how would this outage affect customers?
Long Downtime: Without an SRE team to diagnose and fix the issue, the downtime can be significantly longer. *Bad Customer Experience: Users would face interruptions in service, which could lead to loss of trust in the product.* *Loss of Revenue: For services that are critical to business operations, an outage can result in financial loss both for the customers and for the company providing the service.* *No Communication: A part of the SRE role is to communicate outages and estimated recovery times to stakeholders. Without this, customers wouldn't have visibility.*
4c. What could be put in place so that the SRE team could know of the outage before the customer does?
Monitoring: Tools like Prometheus and Grafana can be used to monitor system health, including API response times, error rates, etc., in real-time. *Automated Alerting: Monitoring tools could be configured to send automated alerts to the SRE team if anomalies like increased error rates or latency are detected.*

Synthetic Transactions: Automated "dummy" transactions could be continuously executed against the API to validate that it's working as expected. Any failures would immediately alert the SRE team.

Log Analysis: Real-time log analysis tools could be employed to search for unusual patterns or errors in the logs, triggering an alert when something is found.

Escalation Policies: Define clear escalation procedures so that if an incident is detected, it immediately goes to the right individuals or teams for quick resolution.

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)?

It received near 5000 bytes at the highest peak shown in the graph.

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

Monitoring Engineer: They are usually the first to know of any incidents and are responsible for managing monitoring dashboards. An increase in traffic can indicate an issue that needs to be closely monitored.

Infrastructure Engineer: If the increase in traffic is high, it could require adjustments to the infrastructure to prevent any disruptions or overloads. The Infrastructure Engineer would need this data to plan for scaling resources or optimizing current usage.

System Architect: While not directly responsible for immediate response, the System Architect would be interested in this data for longer-term planning. They may need to adjust the architecture to account for changes in usage patterns.

Team Lead: As someone who keeps the team focused and is generally aware of all activities, the Team Lead would also be interested in understanding spikes in traffic, especially if it impacts various aspects of the system and requires coordinated effort to address.