

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`

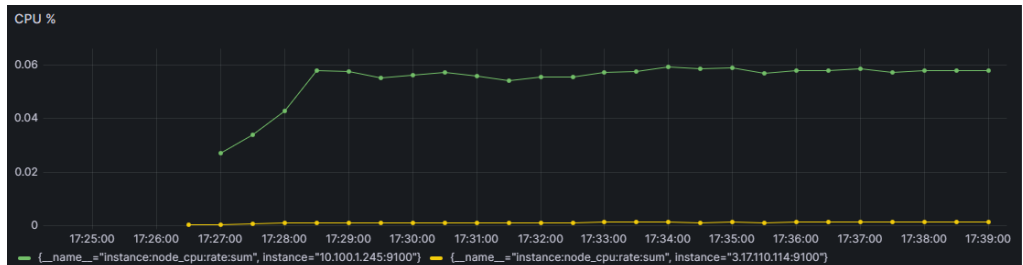
```
Last login: Wed Oct 25 06:28:58 2023 from 196.152.200.79
ubuntu@ip-172-31-31-184:~$ sudo systemctl status node_exporter.service
● node_exporter.service - Node Exporter
   Loaded: loaded (/etc/systemd/system/node_exporter.service; enabled; vendor preset: enabled)
   Active: active (running) since Fri 2023-10-27 15:33:25 UTC; 1min 30s ago
     Main PID: 385 (node_exporter)
        Tasks: 5 (limit: 1088)
       Memory: 139.5M
          CPU: 463ms
    CGroup: /system.slice/node_exporter.service
            └─385 /usr/libexec/qemu-binfmt/x86_64-binfmt-P /usr/local/bin/node_exporter /usr/local/bin/node_exporter

Oct 27 15:33:26 ip-172-31-31-184 node_exporter[385]: level=info ts=2023-10-27T15:33:26.991Z caller=node_exporter.go:115 collector=thermal_zone
Oct 27 15:33:26 ip-172-31-31-184 node_exporter[385]: level=info ts=2023-10-27T15:33:26.991Z caller=node_exporter.go:115 collector=time
Oct 27 15:33:26 ip-172-31-31-184 node_exporter[385]: level=info ts=2023-10-27T15:33:26.991Z caller=node_exporter.go:115 collector=timex
Oct 27 15:33:26 ip-172-31-31-184 node_exporter[385]: level=info ts=2023-10-27T15:33:26.991Z caller=node_exporter.go:115 collector=udp_queues
Oct 27 15:33:26 ip-172-31-31-184 node_exporter[385]: level=info ts=2023-10-27T15:33:26.992Z caller=node_exporter.go:115 collector=uname
Oct 27 15:33:26 ip-172-31-31-184 node_exporter[385]: level=info ts=2023-10-27T15:33:26.992Z caller=node_exporter.go:115 collector=vmstat
Oct 27 15:33:26 ip-172-31-31-184 node_exporter[385]: level=info ts=2023-10-27T15:33:26.992Z caller=node_exporter.go:115 collector=xfs
Oct 27 15:33:26 ip-172-31-31-184 node_exporter[385]: level=info ts=2023-10-27T15:33:26.993Z caller=node_exporter.go:115 collector=zfs
Oct 27 15:33:27 ip-172-31-31-184 node_exporter[385]: level=info ts=2023-10-27T15:33:27.003Z caller=node_exporter.go:199 msg="Listening on" address=:9100
Oct 27 15:33:27 ip-172-31-31-184 node_exporter[385]: level=info ts=2023-10-27T15:33:27.011Z caller=tls_config.go:191 msg="TLS is disabled." http2=false
ubuntu@ip-172-31-31-184:~$ |
```

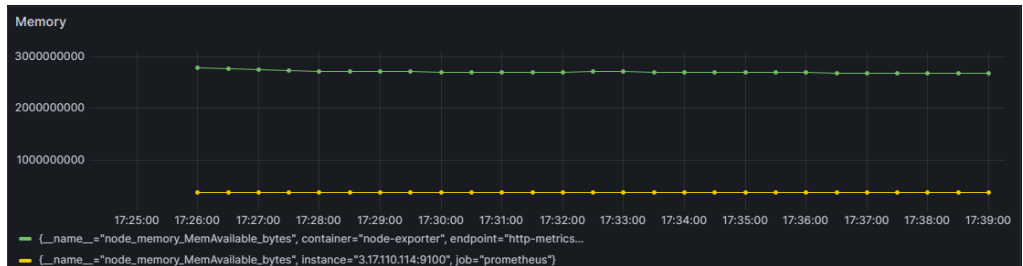
Host Metric (CPU, RAM, Disk, Network)

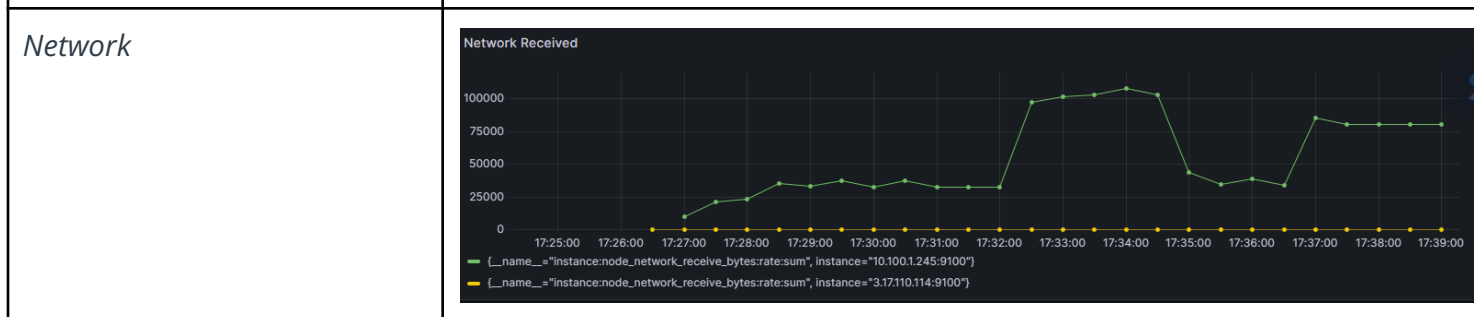
Dashboard

cpu



Memory





Responsibilities	
------------------	--

<p>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.</p>
--

- | |
|---|
| <ul style="list-style-type: none"> - Checking our remaining SLA to make sure we have an error window - Determine which services will be fixed, and make sure this is a small isolated change so nothing else is affected - Using feature flags - Make sure that there is doc for rollback |
|---|

<p>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.</p>

<p><i>To know the system architecture and technologies they gona use, to determine system arch and requirement.</i></p>

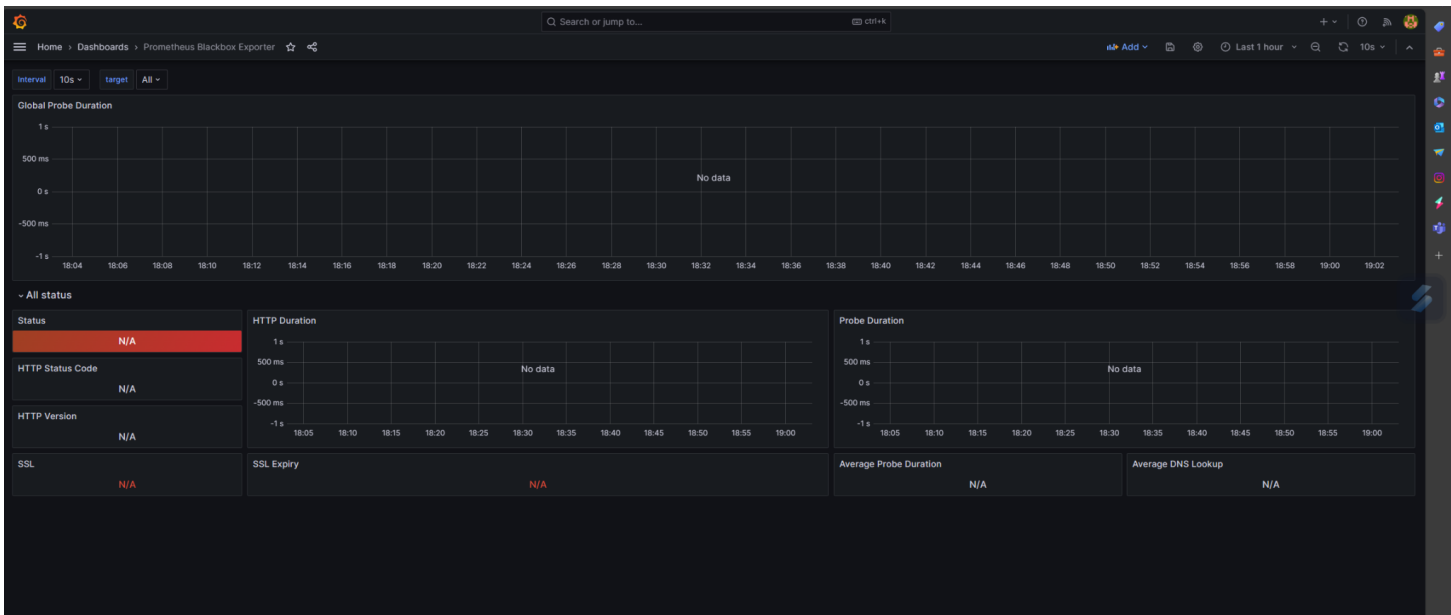
<p>3. The emergency hotfix from question 1 was applied and is causing major issues in production. Which SRE role would primarily be involved in mitigating these issues?</p>
--

<p><i>Infrastructure engineer, he is the one who deploy, and will roll back, and start investigating the problem</i></p>
--

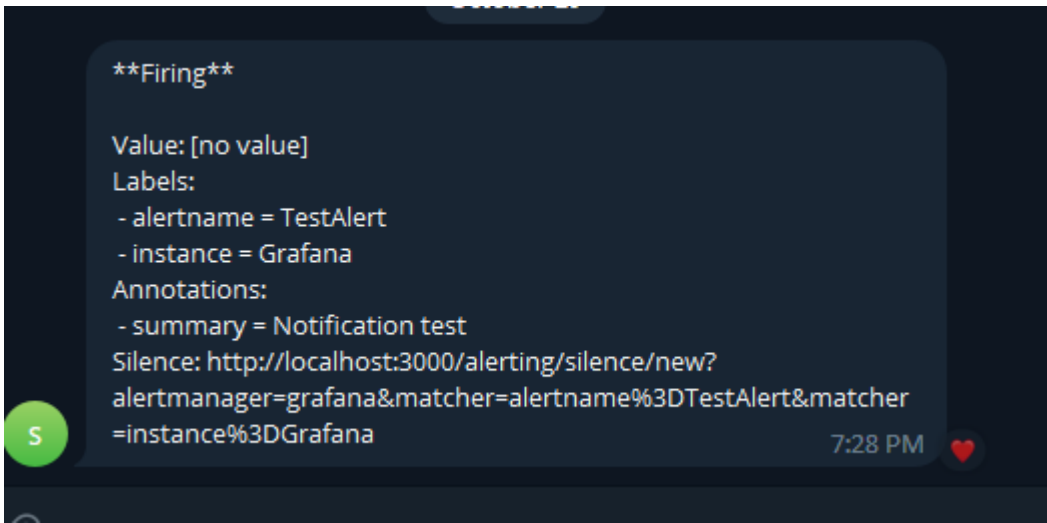
Team Formation and Workflow Identification

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.

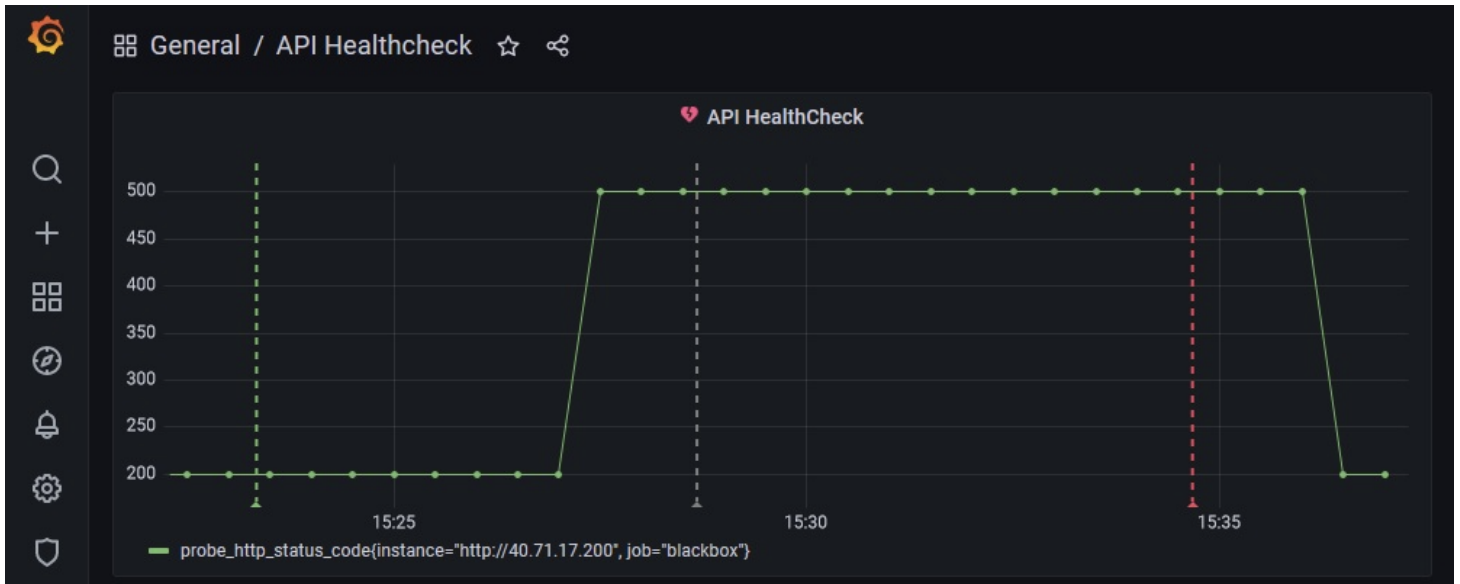


Configure alert rules: Provide a screenshot of the alert rules list in Grafana.

[TODO: copy screenshot here]

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?

15:36

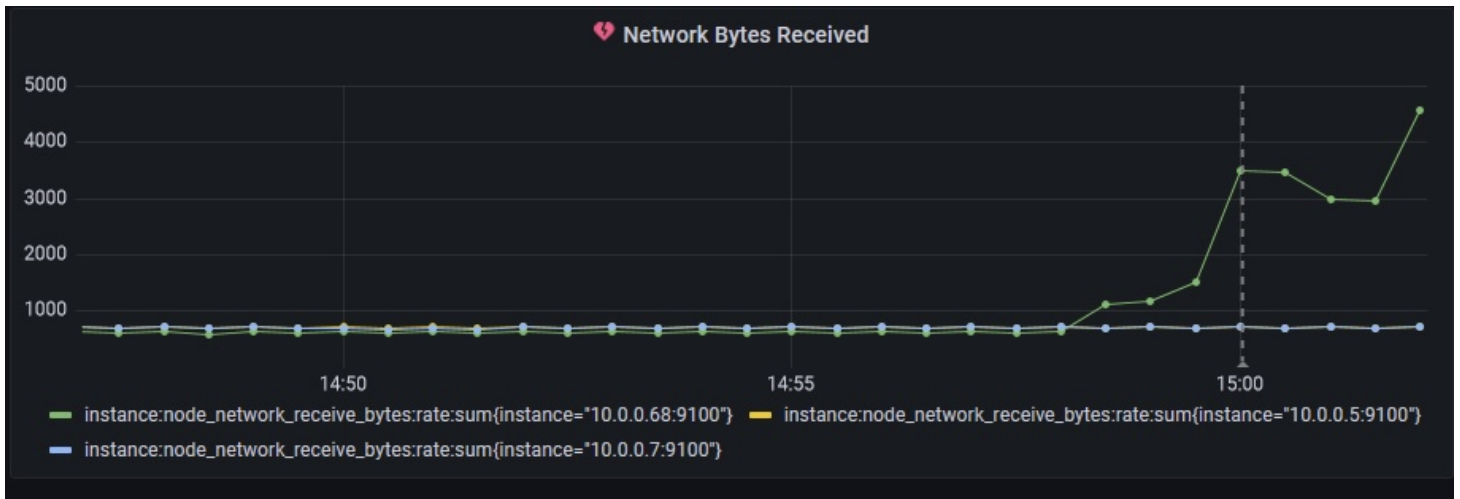
4b. If there was no SRE team, how would this outage affect customers?

No one was going to create this dashboard and alert, so we will know when customers contact us about unreachable website.

4c. What could be put in place so that the SRE team could know of the outage before the customer does?

*Create blackbox with endpoints target, then create alerts if not 2** for 5 time, send notification*

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)?

Instance: 10.0.0.68, and it received almost 5000 bytes (less than 5000, maybe 4800)

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

Monitor Engineer, is the one who makes observation for the system and create dashboards

NOTE:

All values.yaml, blackbox, terraform k8s version, EC2 creation some of the steps are outdated which consumes me a lot of time to figure out what is the problem, also prevents me from assigning in time.

Hope rest of the course not look like that, and been updated.