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 | | |
|  | | |
| **Host Metric**  **(CPU, RAM, Disk, Network)** | **Dashboard** | |
| instance:node\_cpu:rate:sum |  | |
| node\_memory\_MemAvailable\_bytes | *[TODO: copy screenshot of the corresponding metric dashboard here]* | |
| node\_disk\_io\_now |  | |
| instance:node\_network\_receive\_bytes:rate:sum |  | |
| **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 and Monitoring engineer: the release manager will ensure that the rollout is fast but still adheres to the safety playbook rules(it’s a hot fix yes, but it should avoid creating a hot mess), the monitoring engineer will ensure that the hot fix did not cause issues after a successful rollout and additionally could setup extra monitoring alerts/metrics for that hot fix.* | | |
| 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 and SRE Team Lead: the the system architect will help the dev team get off on the right path, and be able to catch bad design/problematic architecture choices, and the team lead, will be there so they are aware of the upcoming responsibilities of their team given the new system and can plan team availability and opportunities for collaboration accordingly. | | |
| 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? | | |
| *Release manager because they are also responsible for rollbacks in case of issues.* | | |

# 

# 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). | | |
| *Cant for the life of me get the black box exporter to work, no metrics are showing up :(((* | | |
| Create a notification channel: Provide a screenshot of the Grafana notification which shows the summary of the issue and when it occurred. | | |
| *[TODO: copy screenshot here]* | | |
| 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? | | |
| *It’s down when the response is 500, from around 15:27:30 -> 15:37:30* | | |
| 4b. If there was no SRE team, how would this outage affect customers? | | |
| *Customers will not be able to use the service, their requests will fail* | | |
| 4c. What could be put in place so that the SRE team could know of the outage before the customer does? | | |
| *We could setup an alert that triggers on too many 500 responses within a time period, say if within any 5 minutes interval if the number of 500 is more than 1% of the total requests, there is an issue and an alert is sent* | | |

| **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, around 5000 bytes* | | |
| 5b. Which team members on the SRE team would be interested in this graph and why? | | |
| *Monitoring engineer is resposable for setting up and monitoring the graphs and setting up the alerts when needed.* | | |

# 

# 