Lab Assignment 3 CMT

Max Shahrokni

TASK 1

- Observability

  ✗ Three pillars of observability

It consists of;
1. Logging - Detailed messages of events from the system.
2. Metrics - This are values representing system resources over time, such as CPU or memory consumption.
3. Tracing - Tracing code from end to end to supervise the flow/latency.

  ✗ Blackbox vs Whitebox monitoring

Blackbox monitoring is testing the application without knowledge of the internal structure or code, an example would be a company testing a finished application for approval. Whitebox monitoring on the other hand, is when the tester do have knowledge of the code. For example software developers doing structural testing of their code.

  ✗ Alerting

Setting triggers to alert administrators of happenings in the system when certain events occur.

  ✗ Monitoring Signals

The monitoring of different types of metrics such as RED, USE and Golden Signals.

  ✗ Why do we need logs?

Logs are records of system events. Leaves an audit trail of who, what, where and why.
There can be many reasons for checking logs, examples being intrusion detection, tracing why things went wrong, debugging and troubleshooting etc.

  ✗ The difference between logs and metrics

Metrics are numbers measured of a given period of time while logs are messages of events that has occured in the system.

  ✗ What is tracing?

Tracing - Tracing code from end to end to supervise the flow/latency.

- Service Mesh

  ✗ What is a service mesh?

Can be described in short as a network between microservices.

- ✗ (Im guessing you mean:) How does it implement its dataplane?

Through service proxies such as envoy.

- ✗ mTLS

Mutual TLS, a two way mutual authentication protocol.

- ✗ Circuit Breaking

A technique that checks flow to an endpoint, if the flow to the endpoint is failing or is to slow, the circuit redirects flow away from the failing endpoint.

- ✗ Traffic Splitting/Steering

To avoid downtime when deploying a new version of a service, this technique is implemented to slowly/gradually shift traffic to the new service.

- ✗ Fault Injection

Fault injection means introducing errors into a system on purpose, used to test the failure recovery capacity of an application.

- ✗ Rolling Update

When updating services, the new version and the old version of pods co-exist, the older version gets scaled down gradually as the new version gets scaled up.

- Serverless

- ✗ Benefits and drawbacks of adopting serverless architecture?

Benefits include;
Economic gain since operational costs are reduced.
Easier to manage since developers need not wrestle with servers, its managed by vendors.
Easier to package and deploy, also faster deployments.
Inherently scalable, the serverless infrastructure will scale automatically as usage increases/decreases.

Disadvantages then;
Testing and debugging can prove more difficult since there is no access to backend processes.
For the same reason(no backend access), there might be unknown security concerns.
Serverless code that lies latent at times may "cold start", causing performance issues.
Can cause vendor lock-in. When relying on a vendor for backend processing, it might prove difficult to switch vendors when necessary.