



Do it once. Do it right. Do it with

**POSEDIO**

Java Meetup  
Vienna

24.06.2024

**POSEDIO**

**Is your spring boot application  
in Kubernetes secure?**



Damjan Gjurovski,  
CTO of Posedio



# IS YOUR SPRING BOOT APP SECURE

1. Ever heard of CIA?
2. Kubernetes == Availability?
3. Hey, this is not my user!
4. Where is all that data going?

# HI

- Damjan Gjurovski
- Java & Kubernetes fan
- Had to secure my own java applications on k8s, and then had to secure other peoples java applications on k8s

How to be good at talking

1. Polite greeting
2. Name
3. Relevant personal link
4. Manage expectations





Ever heard of CIA?

# CIA TRIAD

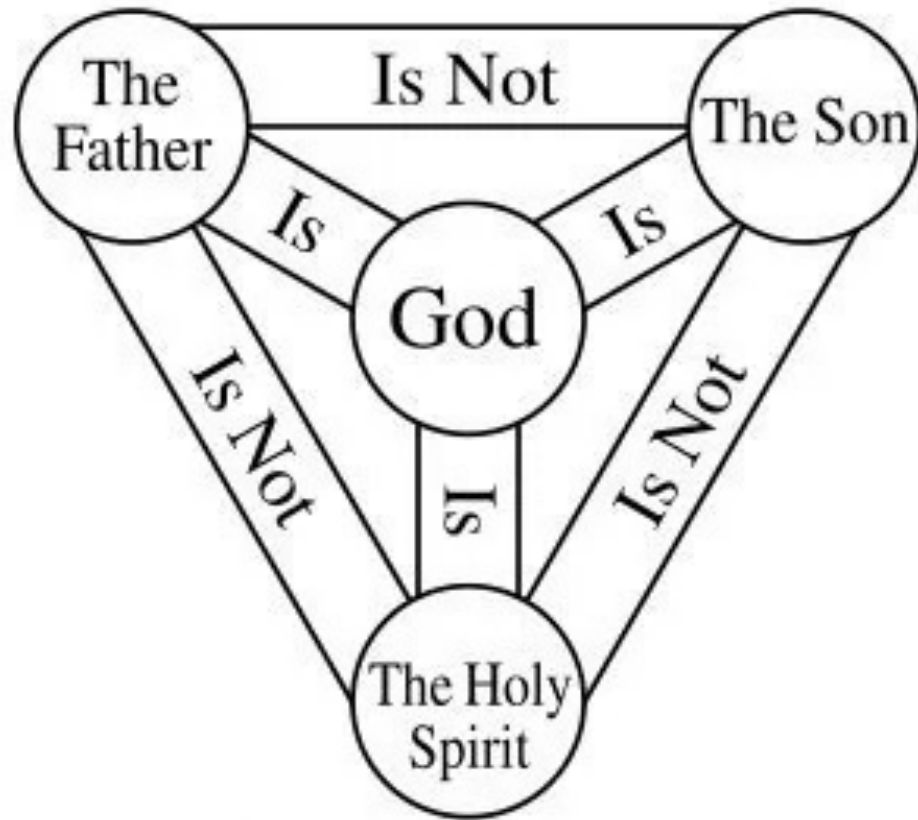
Confidentiality

Integrity

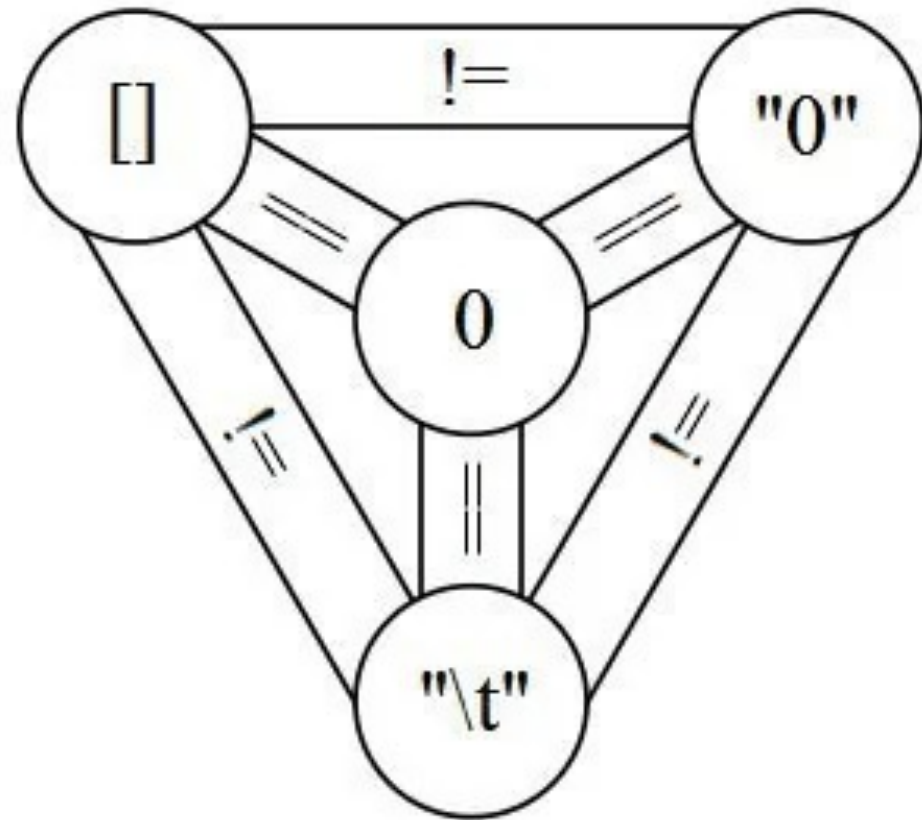
Availability



## CIA TRIAD



Christianity



JavaScript

@hsjoins



# Availability

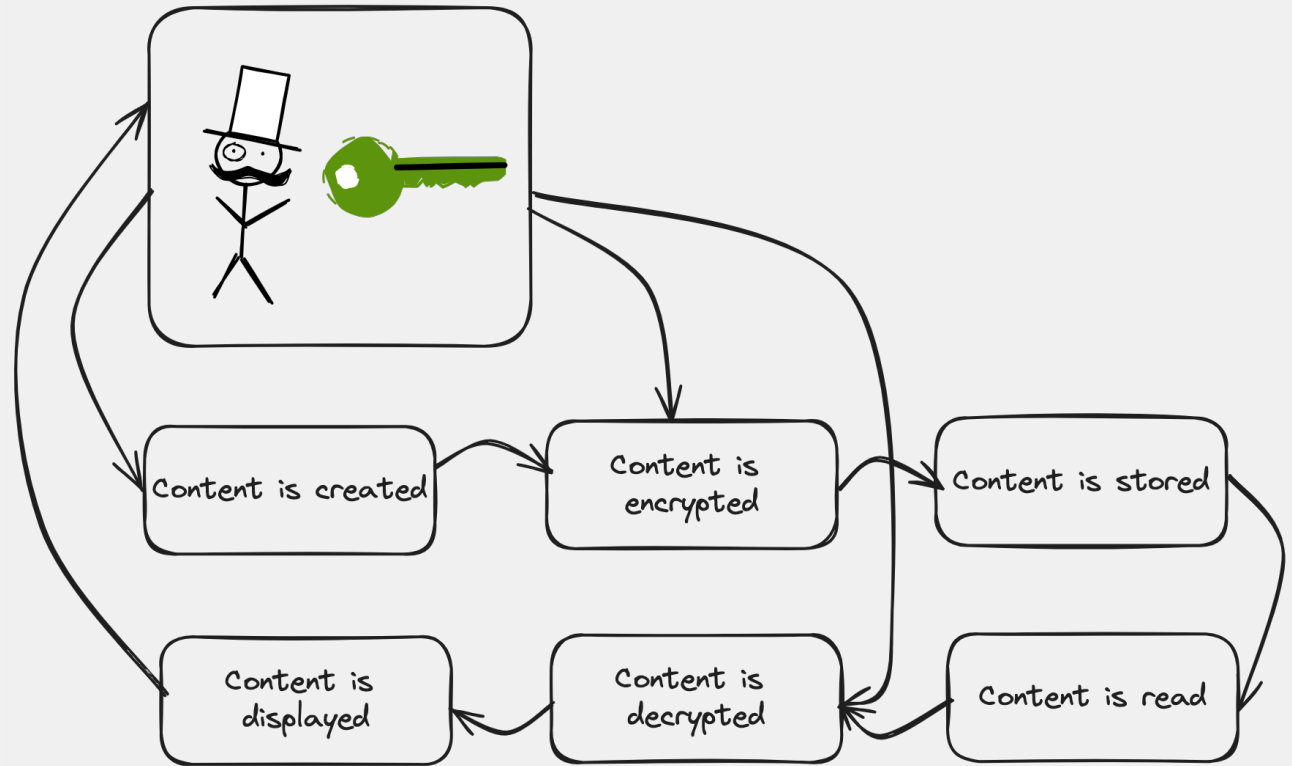


# AVAILABILITY

Your application needs to be accessible (available) to be useful

The main question is: Can I access my data when I need it?

Means we care not only for uptime but also latency

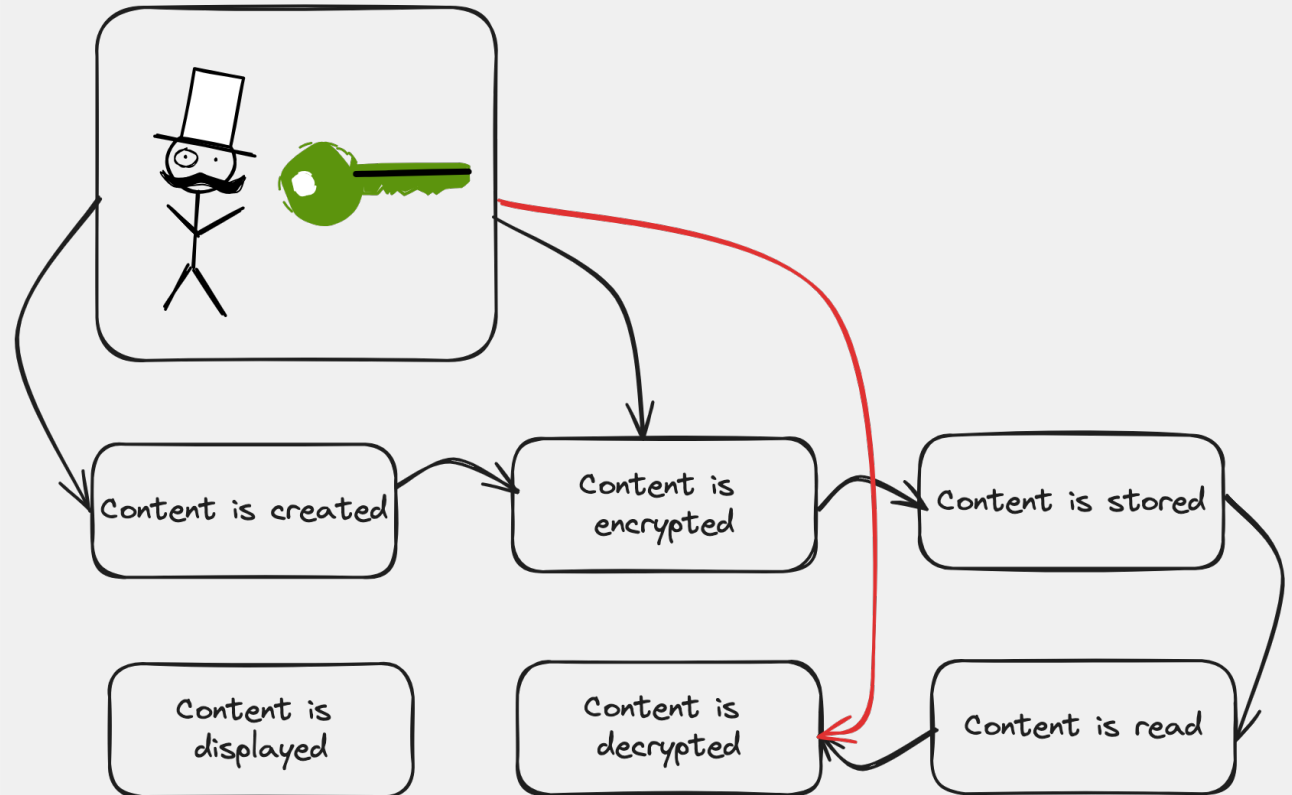


# AVAILABILITY

Your application needs to be accessible (available) to be useful

The main question is: Can I access my data when I need it?

Means we care not only for uptime but also latency

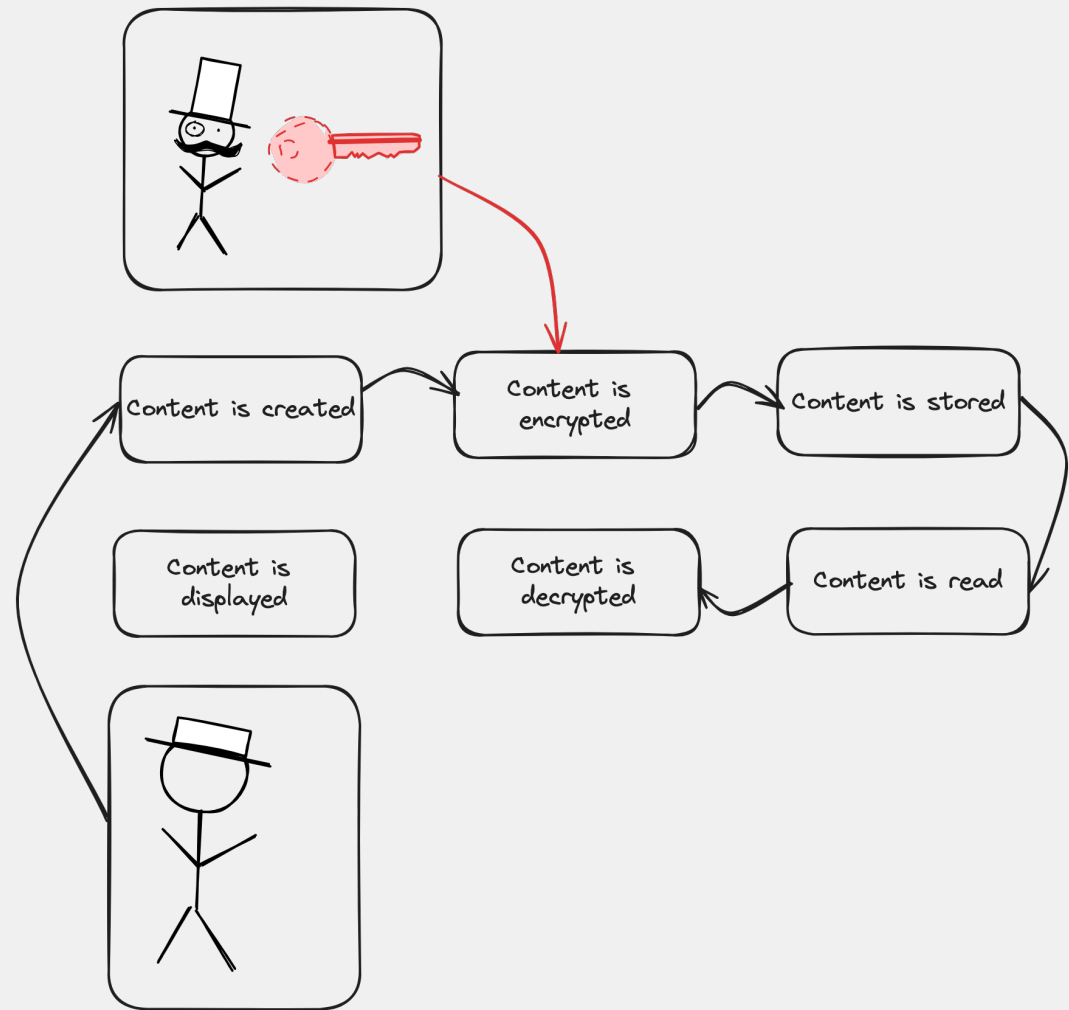


# AVAILABILITY

Your application needs to be accessible (available) to be useful

The main question is: Can I access my data when I need it?

Means we care not only for uptime but also latency



# KUBERNETES != AVAILABILITY

Kubernetes cares about uptime and application health, not security

K8s will restart your app if it thinks its not healthy! But how does it know?

The infamous CrashLoopBackoff

```
livenessProbe:
  httpGet:
    path: "/actuator/health/liveness"
    port: <actuator-port>
  failureThreshold: ...
  periodSeconds: ...

readinessProbe:
  httpGet:
    path: "/actuator/health/readiness"
    port: <actuator-port>
  failureThreshold: ...
  periodSeconds: ...
```

# THUNDERING HERD

Self-inflicted DoS when many requests wait for an event and then all fire at once

Easy to happen with readiness probes

Problem gets compounded by restarts

Fail-open mode adds more load

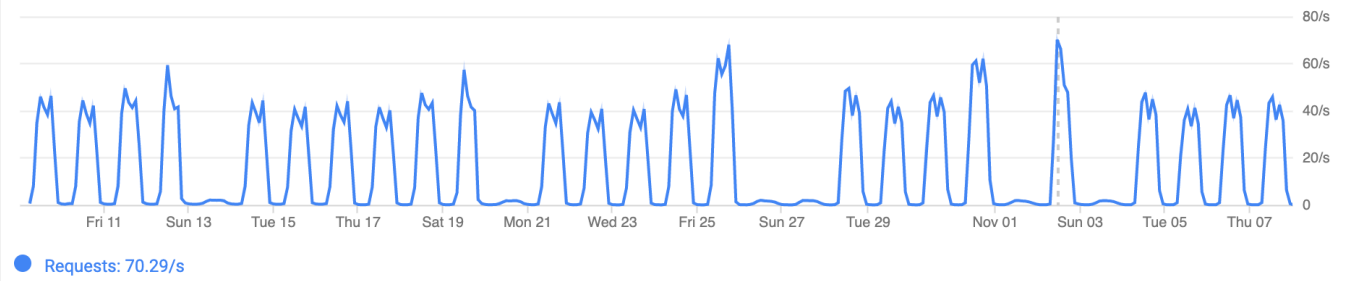
Include backoff and jitter in your @Retryable

## Requests

Requests/sec (6 hr average)

Nov 2, 2019 11:33 AM

0 interval



# ACCESS FROM THE OUTSIDE

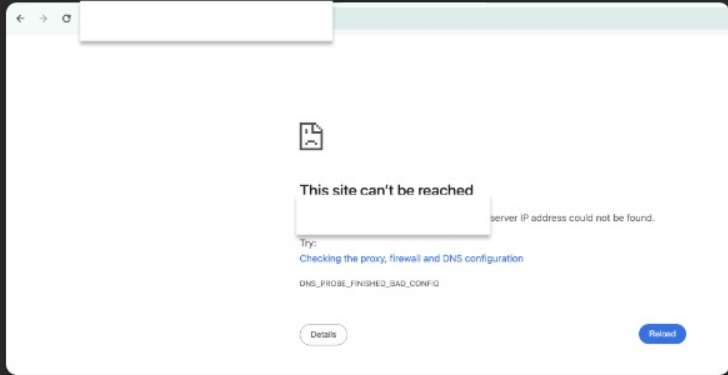
All the clever tricks in Kubernetes focus on your application as viewed by the cluster

What counts is if the user can access the application

User (Developer)

DEV ArgoCD down?

Hello Application Platform I can't access Service but staging and prod seem to be fine.



User (Developer)

same here

User (Developer)

works for me, but is slow as hell

Platform Engineer

the uptime check in our monitoring looks ok and I can also reach it. We will investigate why is it slow.

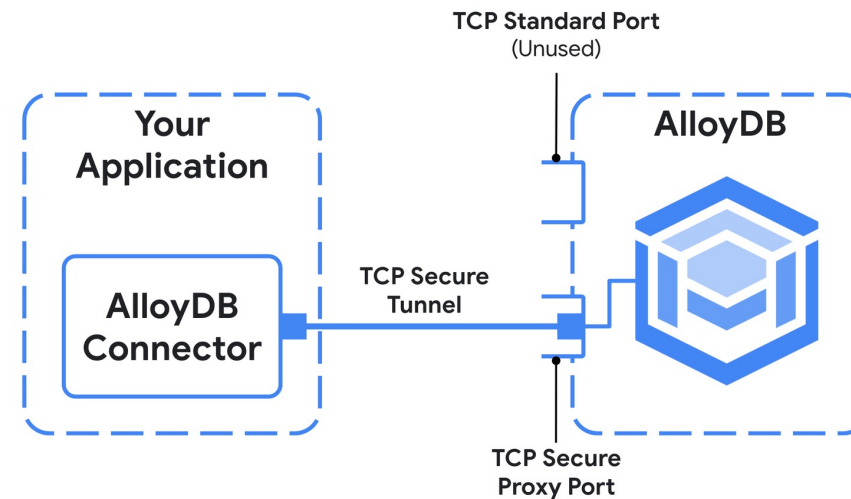
# DOWNSTREAM HEALTH CHECKS

Spring allows you to incorporate downstream services in your health checks

Useful if you want to know if the database is available

But expensive when you make network calls

Very expensive if you perform computations there





Integrity



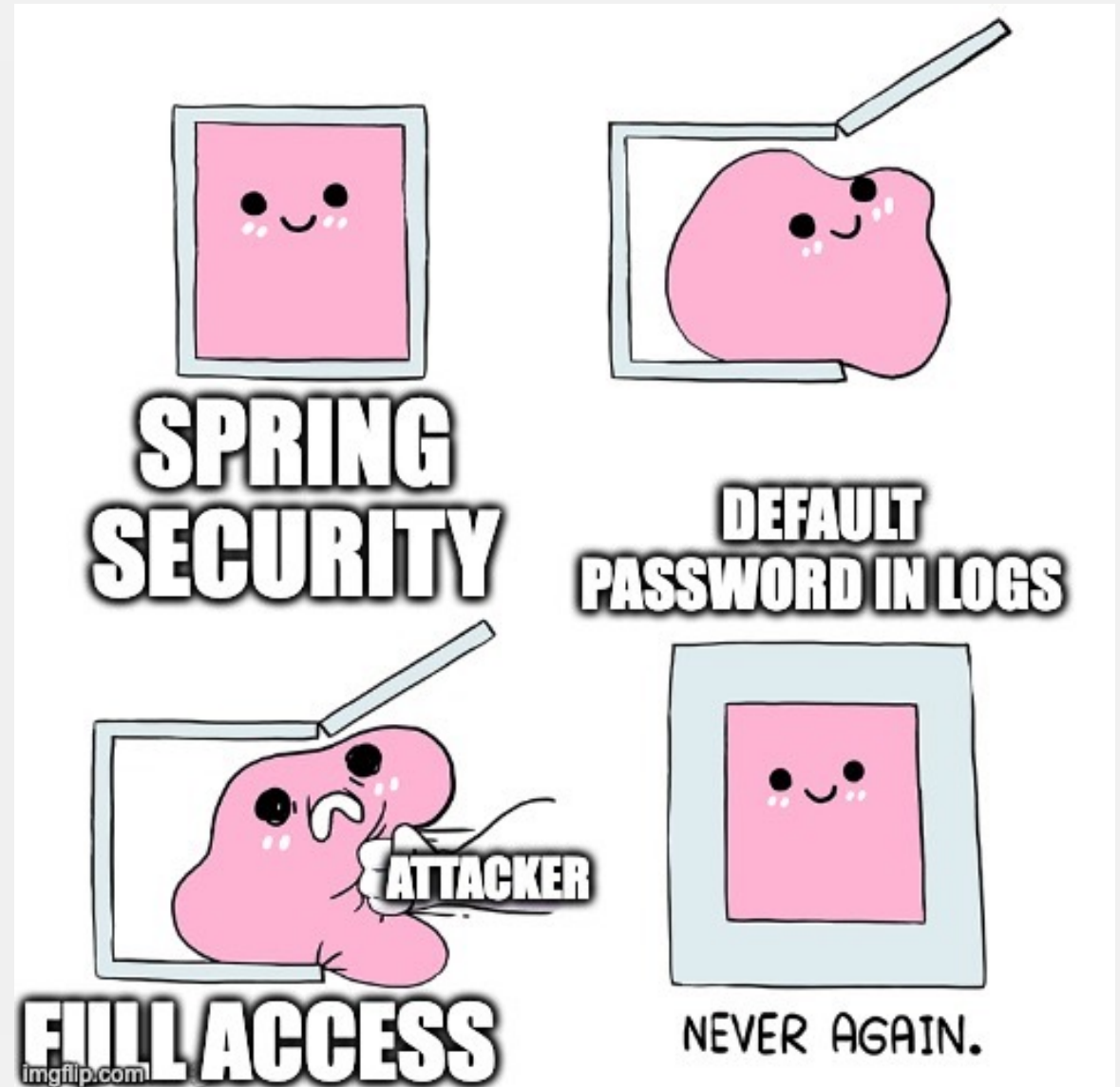
# INTEGRITY

Integrity is about protecting data against unauthorized modification and assuring data trustworthiness.

Data integrity - data has not been changed accidentally or deliberately

Source integrity - data came from or was changed by a legitimate source

Spring Security is usually a good way to handle Authn/Authz and thus ensure data integrity



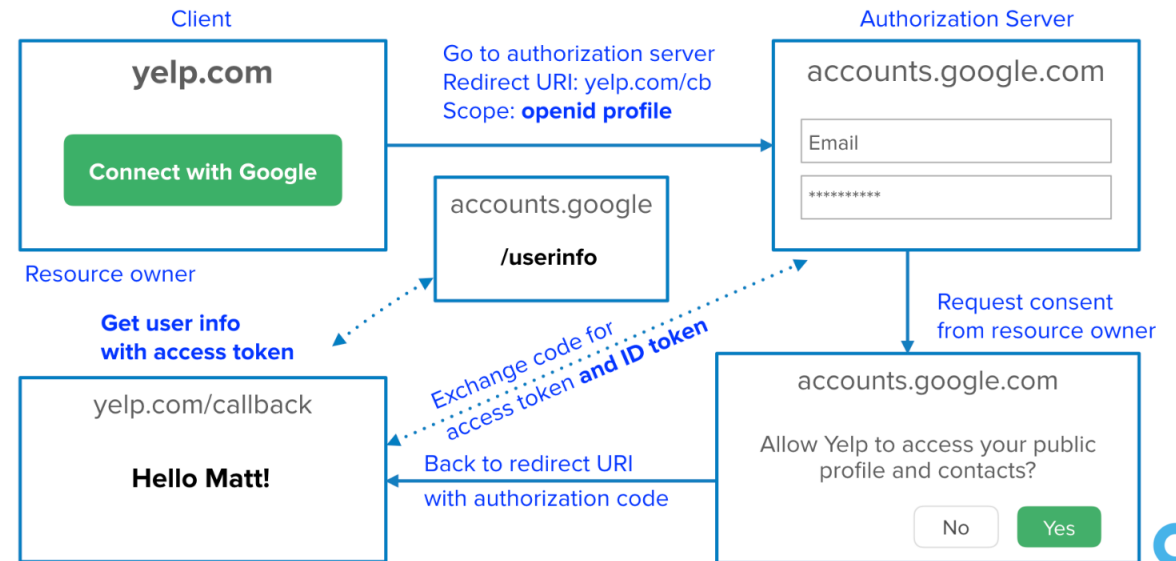
# OIDC WITH SPRING

Complex setup with authorization tokens, access tokens and refresh tokens, PKCE

JWT expiry, revocation and propagation

Should access to the health endpoints be behind authorization?

## OIDC Authorization Code Flow



# IMMUTABLE IMAGES

Tags are not immutable!

Kubernetes will by default pull the latest image, unless its already present on the machine -

pullPolicy: ifNotPresent

This means you can have different versions of the image on different machines, depending on when they were pulled!

## Released ESP docker images

ESP docker images are released regularly. The regular images are named as `gcr.io/endpoints-release/endpoints-runtime:MAJOR_VERSION.MINOR_VERSION.PATCH_NUMBER`. For example, `gcr.io/endpoints-release/endpoints-runtime:1.30.0` has `MAJOR_VERSION=1`, `MINOR_VERSION=30` and `PATCH_NUMBER=0`.

Symbolically linked images:

- **MAJOR\_VERSION** is linked to the latest image with same **MAJOR\_VERSION**.

For example, `gcr.io/endpoints-release/endpoints-runtime:1` is always pointed to the latest image with "1" major version.

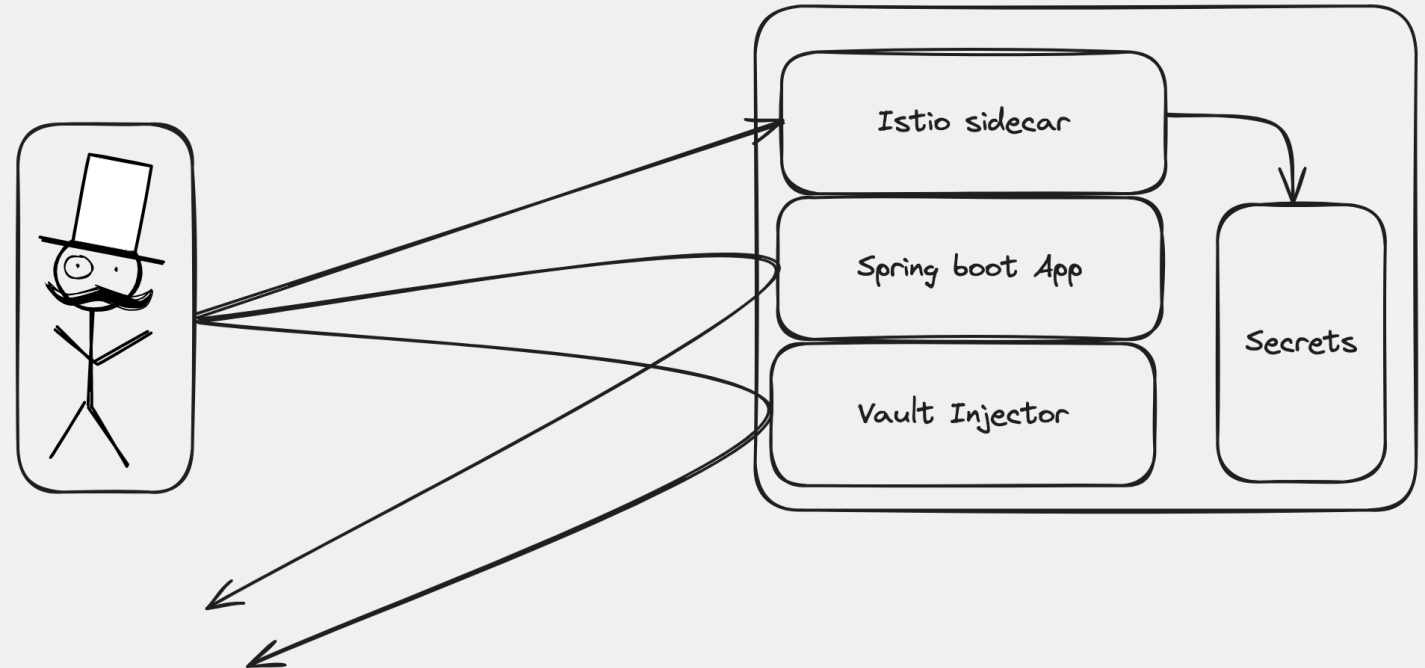
# DATA ON DISK

Where is the disk coming from?

Containers have their own filesystem, but pods share a filesystem

Containers can write to the disk of the machine

Persistent Volumes can attach disks to a pod – cleanup is not always guaranteed!





# Confidentiality

# CONFIDENTIALITY

Secret data should stay secret!

Or, more formally, only people with the correct authorization can access protected data



# WHO HAS ACCESS TO YOUR LOGS

Kubernetes does not handle logging out of the box nicely

Many tools can be used to collect logs, not all of them behave equally

Access to logs is not always restricted



# KUBERNETES PRIVILEGES

Running in privileged mode can give people a lot of access

You have no control if other users run in privileged mode

## Warning:

Any containers that run with `privileged: true` on a node can access all Secrets used on that node.

The `emptyDir.medium` field controls where `emptyDir` volumes are stored. By default `emptyDir` volumes are stored on whatever medium that backs the node such as disk, SSD, or network storage, depending on your environment. If you set the `emptyDir.medium` field to `"Memory"`, Kubernetes mounts a tmpfs (RAM-backed filesystem) for you instead. While tmpfs is very fast be aware that, unlike disks, files you write count against the memory limit of the container that wrote them.

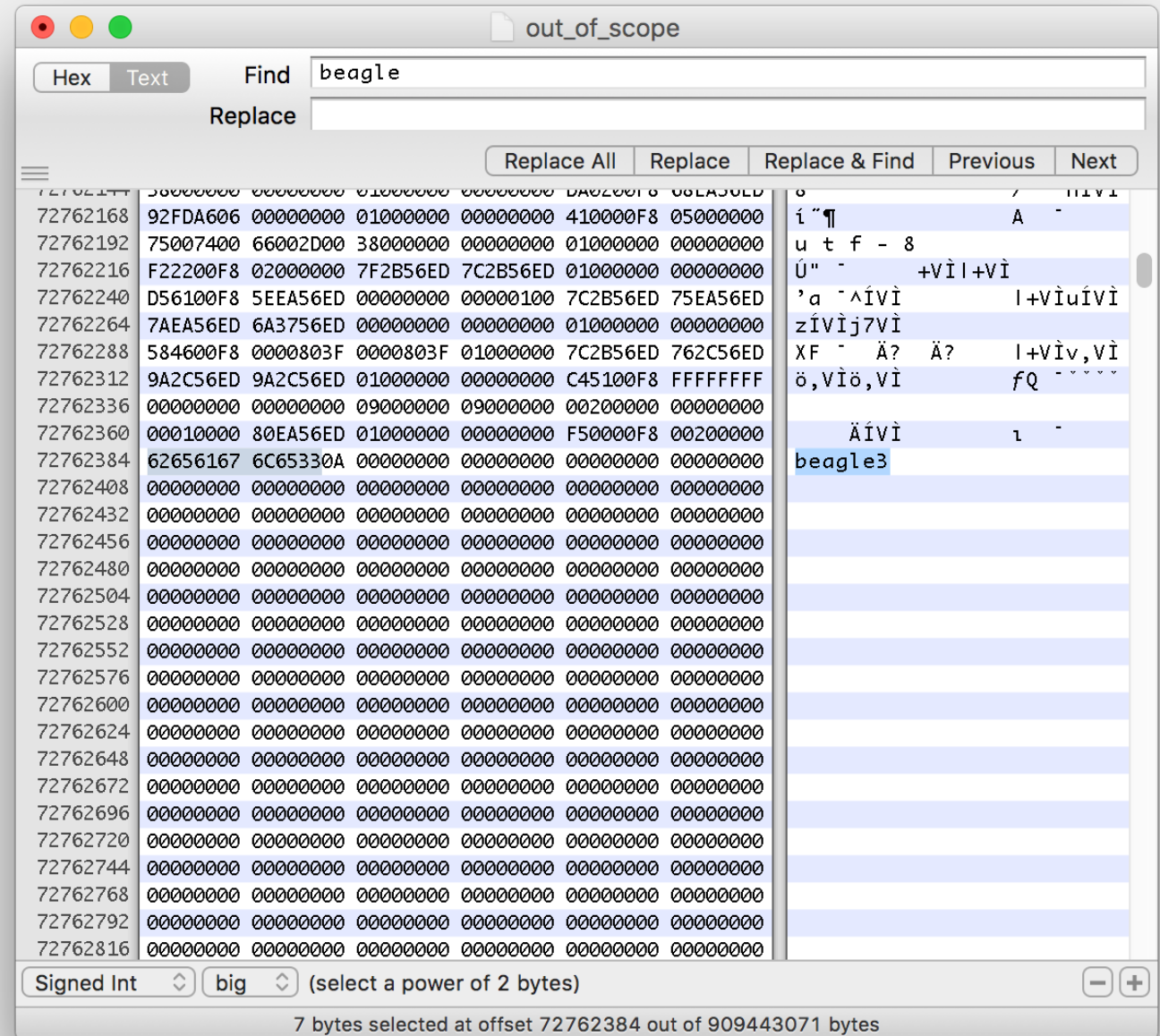


# SPRING CONFIG

Mounting configmaps can be great to switch spring profiles between environments

Configmaps can enable the actuator endpoint

Actuator endpoint exposes the heap, and therefore potentially passwords stored in memory!





Maximum security

# THIS IS THE WAY

Understand the runtime

Take responsibility

Look for attack vectors

Follow best practice!

