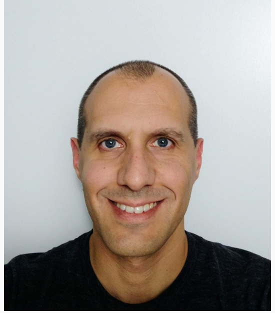
Visualize your Chaos Engineering experiments with Grafana

Intro



Christophe Fargette
Co founder Phlyt & Cloud Native Developer

Phlyt Inc.
christophe@phlyt.io
www.phlyt.io

Github: christophe-f
Twitter: _christophe_f

Agenda

- What is Chaos Engineering?
- How to inject failures safely?
- Type of chaos
- Review Architecture
- Demo
  - Experiment 1 - Inject Latency
  - Experiment 2 - Inject Exceptions
- Questions

What is Chaos Engineering?

- Art of breaking things in purpose
- Prevent/minimise downtime
- Reproduce outage
- Test new infrastructure
- Test partial deleting Kafka topics
- DNS unavailability
- Random I/O errors
- Maxing out CPU cores

Best practices. How to experiment safely?

- Having a resilient system
- Monitoring & alerts
- Start in non prod environment
- Plan your experiment (small scope)
- Have a rollback plan
- Communication
- & Communication

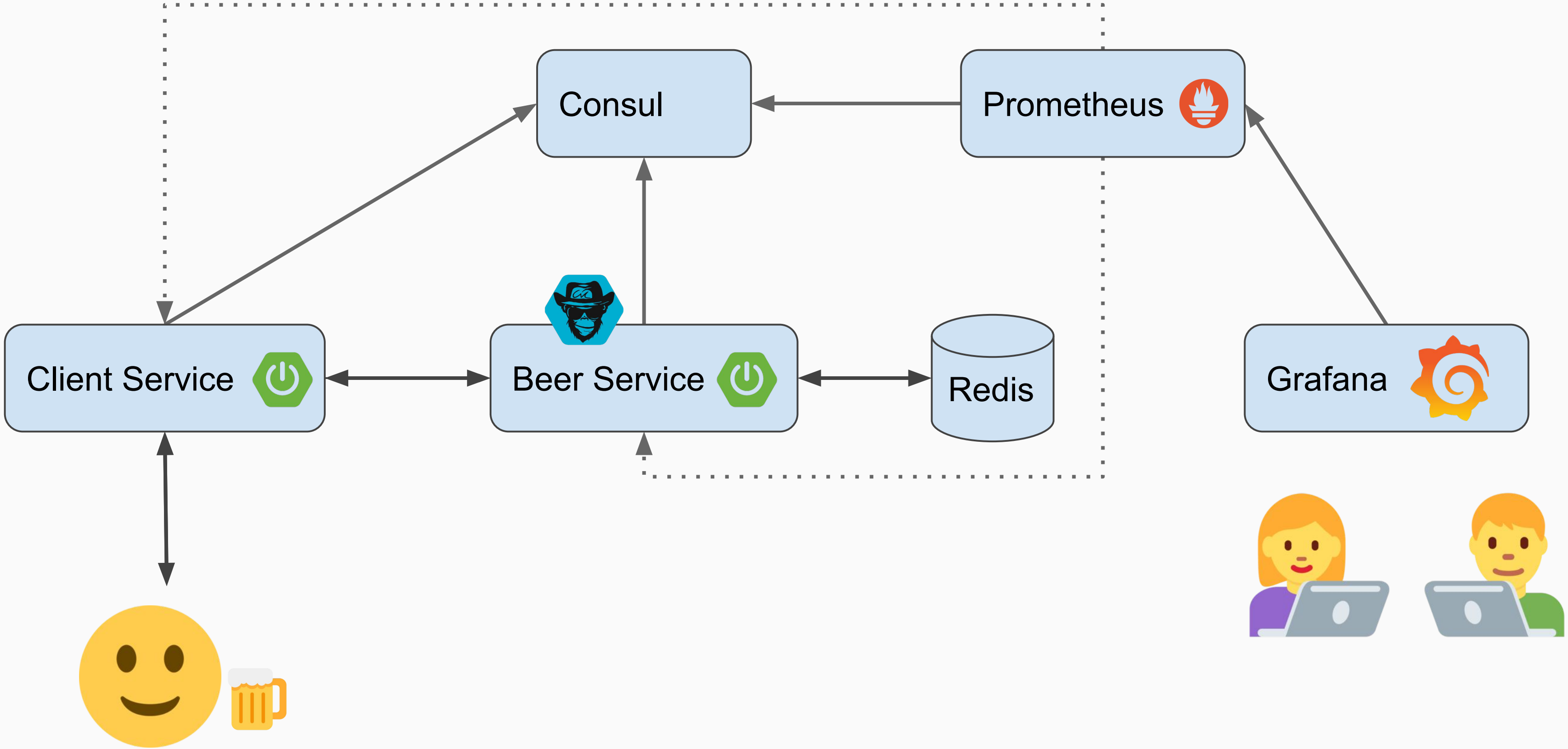
Types of chaos

- Chaos Monkey (Instances)
- Chaos Lemur (Bosh VMs)
- Chaos Gorilla (AZ)
- Chaos Kong (Region)
- Security Monkey
- More ...

Tooling

- Chaos Monkey for Spring Boot
- Chaos Toolkit (https://chaostoolkit.org/)
- Gremlin (https://www.gremlin.com/)
- More ...

Setup


Consul
Prometheus
Client Service
Beer Service
Redis
Grafana

Demo

Experiment 1 - Inject latency

- Test the system under normal circumstance
- Inject latency into Beer Service
- Make sure client service is not timing out
- Fix it if the experiment fails

Experiment 2 - Inject exceptions

- Test the system under normal circumstance
- Inject exceptions into Beer Service
- Make sure client service is resilient
- Fix it if the experiment fails

Are you ready for chaos
in production yet?

Thanks!

Phlyt Inc.
christophe@phlyt.io
www.phlyt.io

Github: christophe-f
Twitter: _christophe_f

Github Demo:
https://github.com/christophe-f/chaos-monkey

> “It’s helpful to think of a vaccine or a flu shot. While seemingly counterintuitive, you inject yourself with something harmful in order to prevent a future issue”

- From Kolton Andrus, CEO of Gremlin