Chaos in the Cloud

Bent Krause (he/him)

Solutions Architect
Amazon Web Services

Oliver Steenbuck (he/him)

Solutions Architect
Amazon Web Services

Agenda

- Recap: What is Chaos Engineering?
- Chaos engineering in the Cloud
  - What is the cloud?
- AWS Fault Injection Simulator
- Demo
- Best practices

What is chaos engineering ?

> **Failures are a given and everything will eventually fail over time.**

Werner Vogels
CTO – Amazon.com

“Chaos Engineering is the discipline of experimenting on a distributed system in order to build confidence in the system’s capability to withstand turbulent conditions in production.”

https://principlesofchaos.org

STRESS

OBSERVE

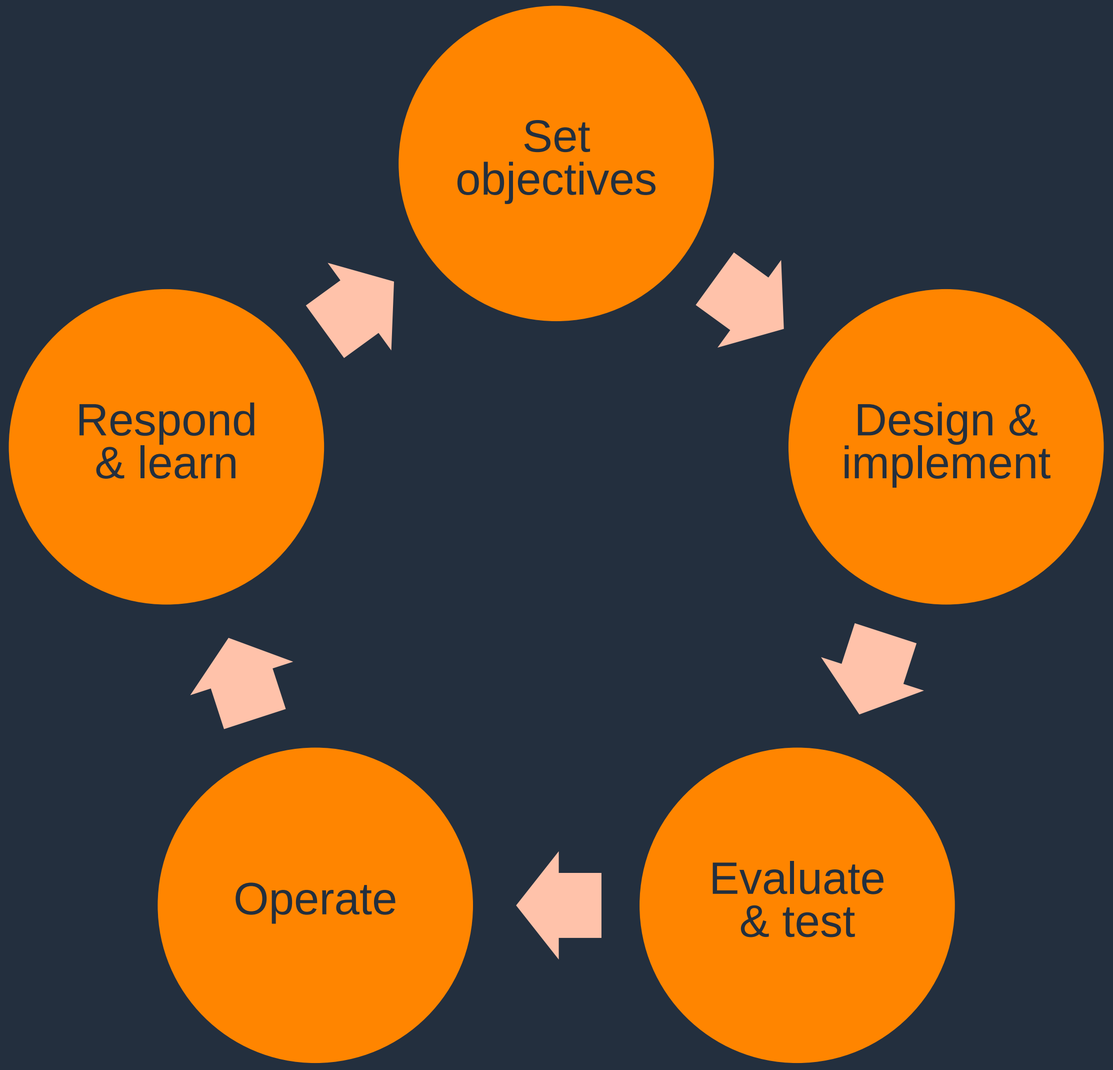
IMPROVE

Chaos engineering

- ✓ Improve resilience and performance
- ✓ Uncover hidden issues
- ✓ Expose blind spots
  Monitoring, observability, and alarm
- ✓ And more

New! The AWS resilience lifecycle framework

aws.amazon.com/resilience



Set objectives

Design & implement

Evaluate & test

Operate

Respond & learn

Chaos engineering
JVM vs. Cloud

Java Chaos Engineering

- JVM based
- Agent based
- Service Mesh based

- Test code
- Control experiment from application code

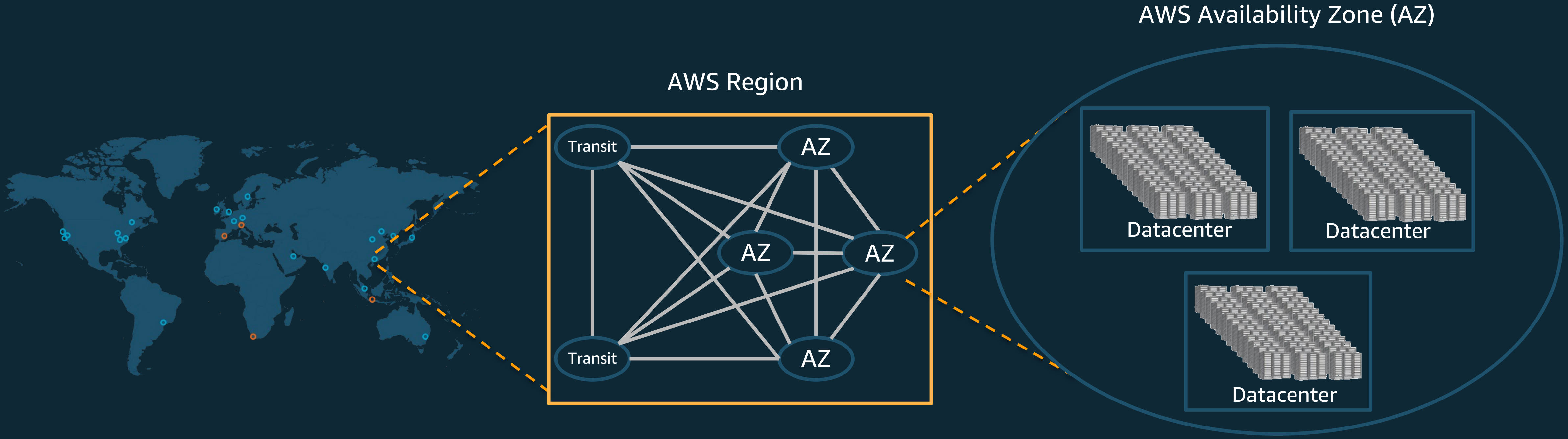
Chaos Engineering in the Cloud

- AWS Fault Injection Simulator

- Holistic Monitoring
- Identification of cloud based challenges
- Scaling experiments
- Validation of disaster recovery

AWS Region design

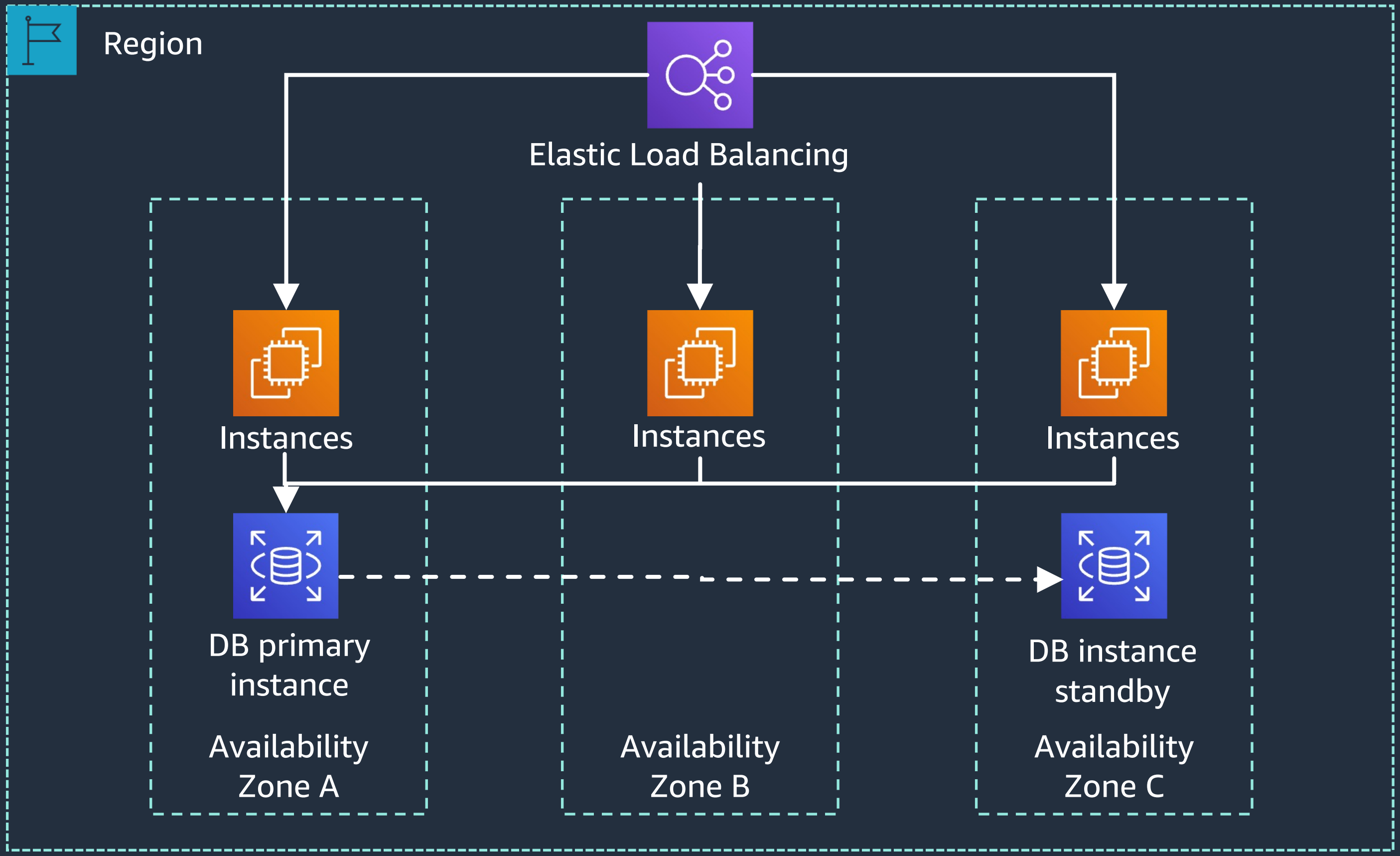
AWS Regions are comprised of multiple AZs for high availability, high scalability, and high fault tolerance. Applications and data are replicated in real time and consistent in the different AZs.



A Region is a physical location in the world where we have multiple **Availability Zones.**

Availability Zones consist of one or more discrete data centers, each with redundant power, networking, and connectivity, housed in separate facilities.

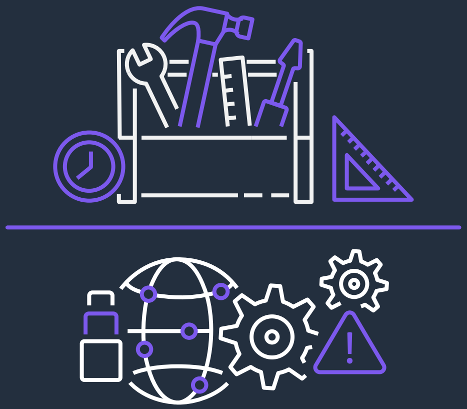
Multi-AZ application



AWS Fault Injection Simulator

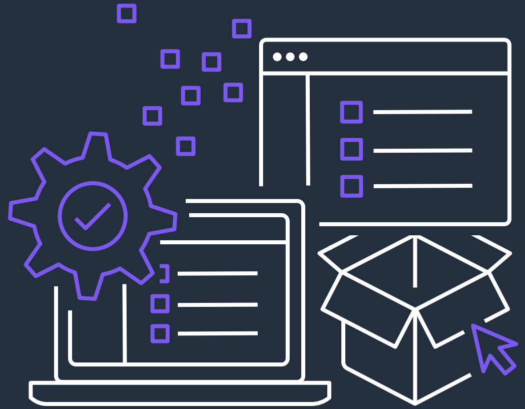
Fully managed chaos engineering service

Easy to
get started



Real-world
conditions

Safeguards



Easy to get started

- No need to integrate multiple tools and homemade scripts or install agents
- Use the AWS Management Console or the AWS CLI
- Use pre-existing experiment templates and get started in minutes
- Easily share it with others

Real-world conditions

- Run experiments in sequence of events or in parallel
- Target all levels of the system (host, infrastructure, network, etc.)
- Real faults injected at the service control plane level!

Safeguards

- “Stop conditions” alarms
- Integration with Amazon CloudWatch
- Built-in rollbacks
- Fine-grain IAM controls

Native Targets And Actions

Compute

- Amazon Elastic Compute Cloud (Amazon EC2)
- Amazon Elastic Kubernetes Service (Amazon EKS)
- Amazon Elastic Container Service (Amazon ECS)

Storage

- Amazon Simple Storage Service (Amazon S3)
- Amazon Elastic Block Store (Amazon EBS)

Networking

- Amazon Virtual Private Cloud (Amazon VPC)

Database

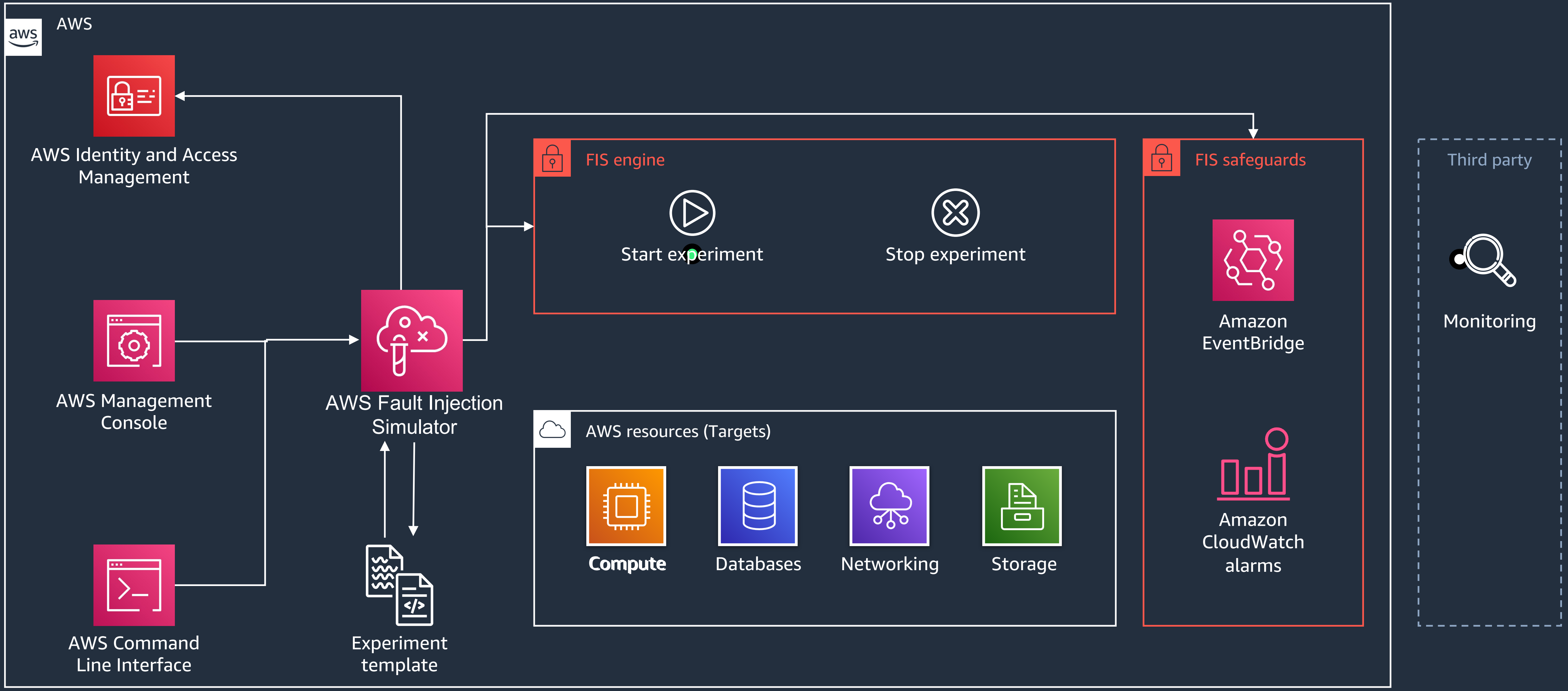
- Amazon Relational Database Service (Amazon RDS)
- Amazon DynamoDB

Management

- Amazon CloudWatch
- AWS Systems Manager

AWS Fault Injection Simulator – Reference Flow

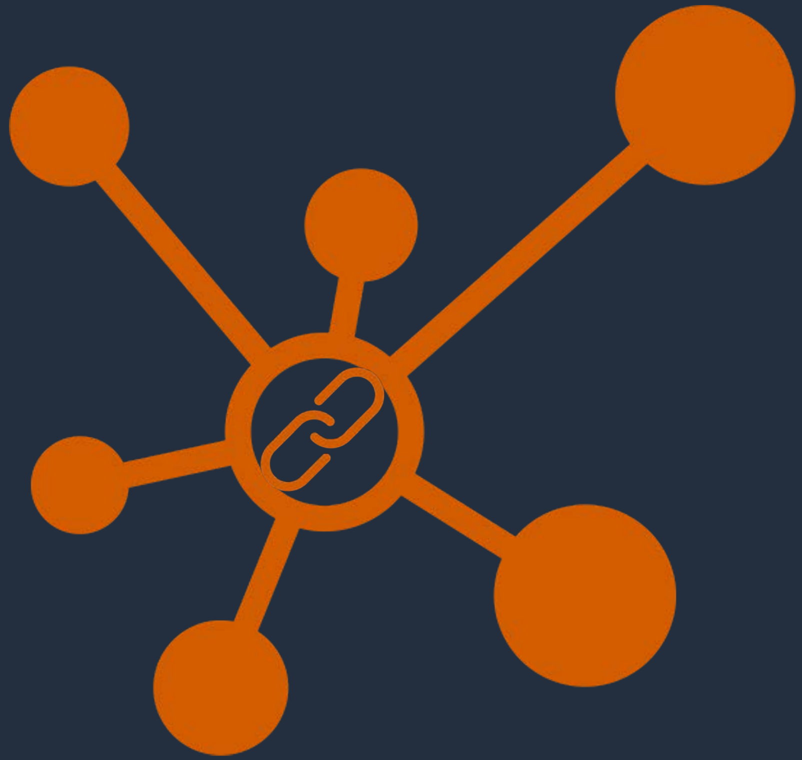
OVERVIEW



Announcing two new scenarios!

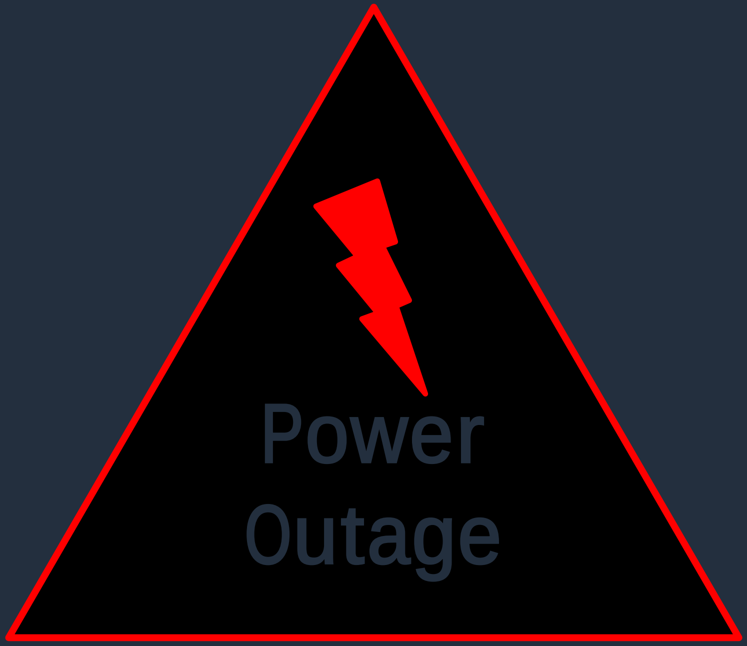
Cross-Region: Connectivity

Test multi-Region applications can operate when their primary Region is inaccessible



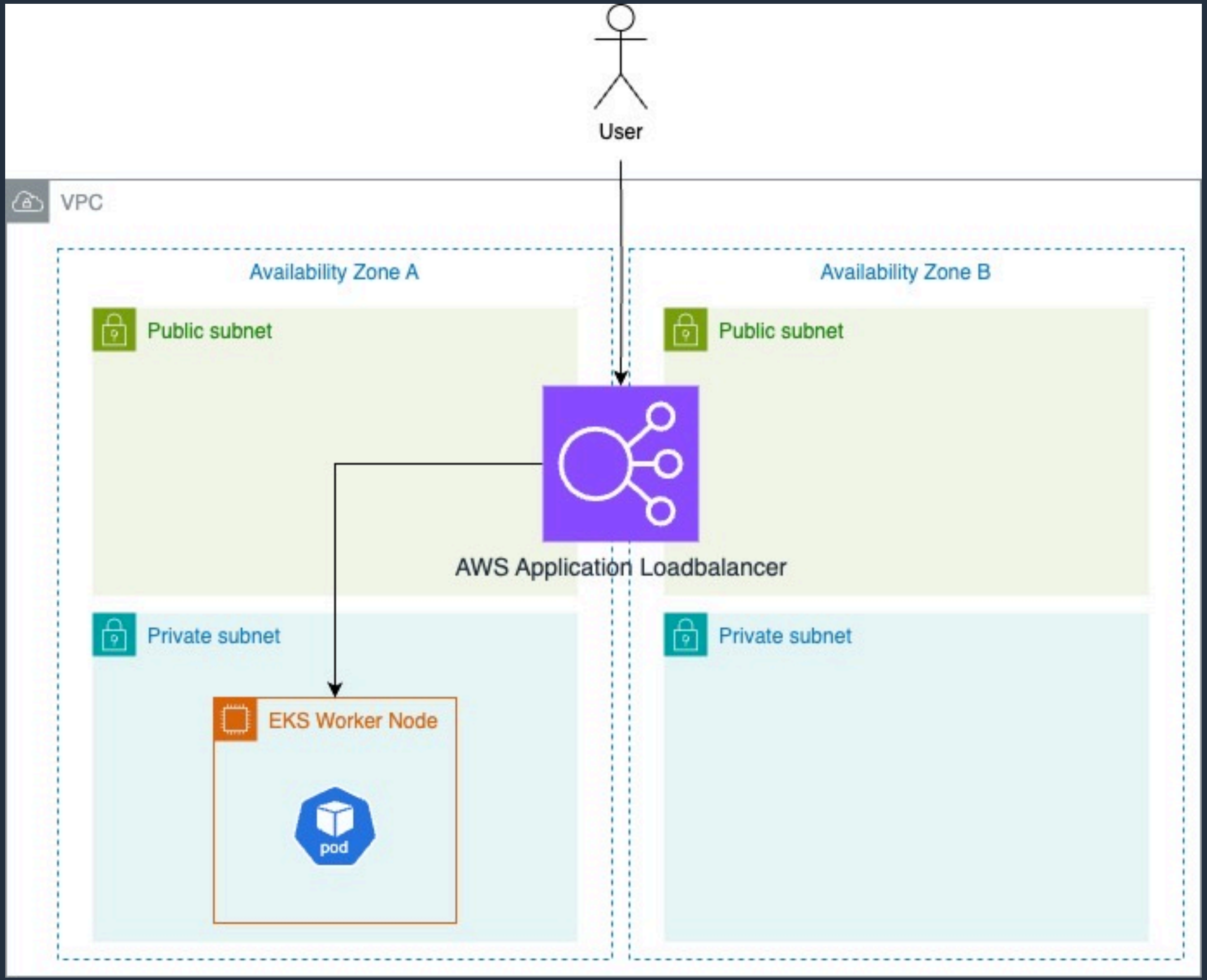
AZ Availability: Power Interruption

Test a Multi-AZ application against the most common effects of a complete AZ power interruption

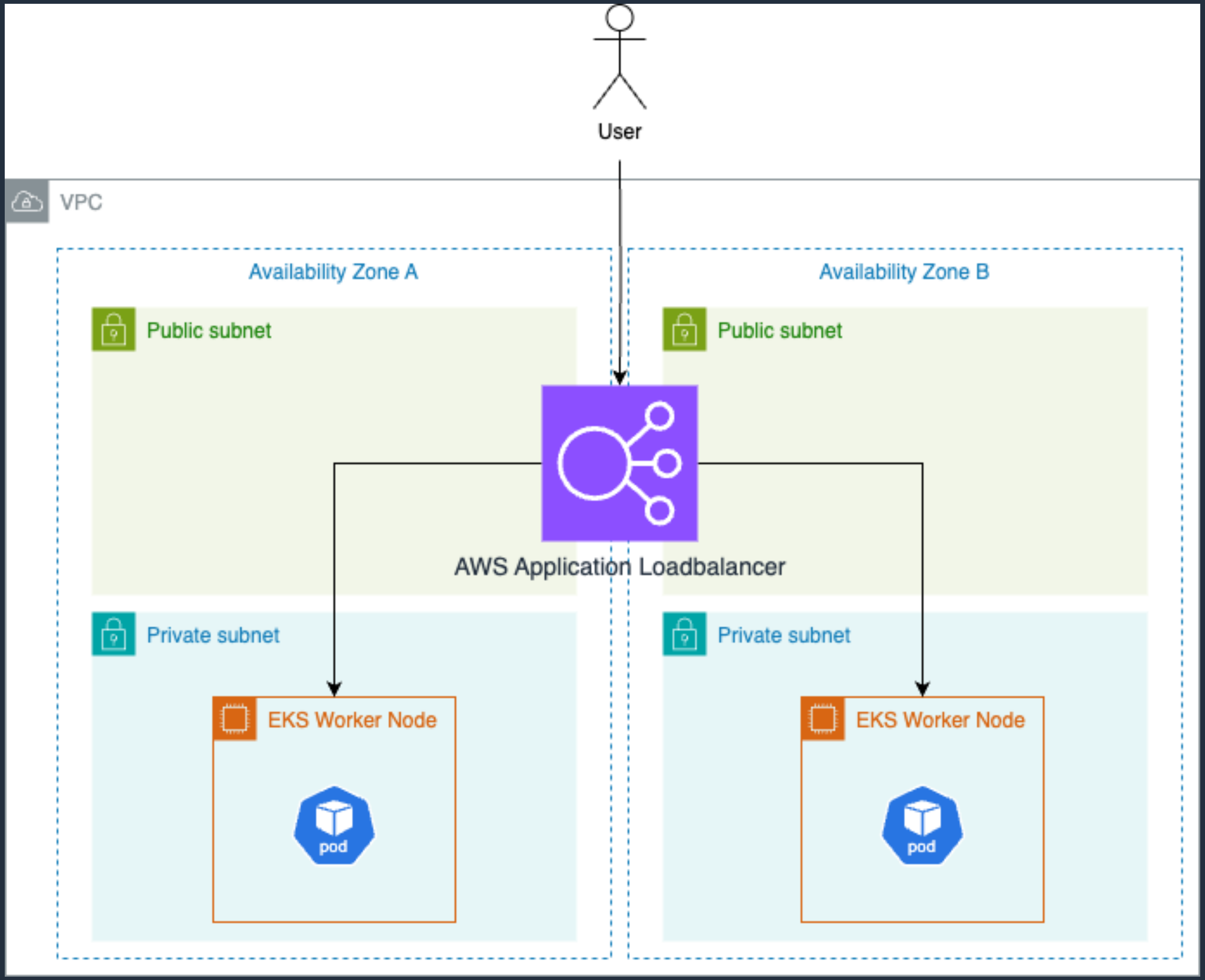


Demo

Architecture diagram one AZ



Architecture diagram two AZs



Rules of thumbs

- Start with very small experiment
- As close as possible to production
- Minimize the blast radius
- Have an emergency STOP!

AWS Fault Injection Simulator

https://catalog.us-east-1.prod.workshops.aws/workshops/5fc0039f-9f15-47f8-aff0-09dc7b1779ee/en-US

Thank you!

Bent Krause (he/him)
Solutions Architect
Amazon Web Services

Oliver Steenbuck (he/him)
Solutions Architect
Amazon Web Services

https://pulse.aws/survey/CTZW4ZTH