



# Continuous Reliability.

## How?

Ana Margarita Medina



[@ana\\_m\\_medina](https://twitter.com/ana_m_medina)

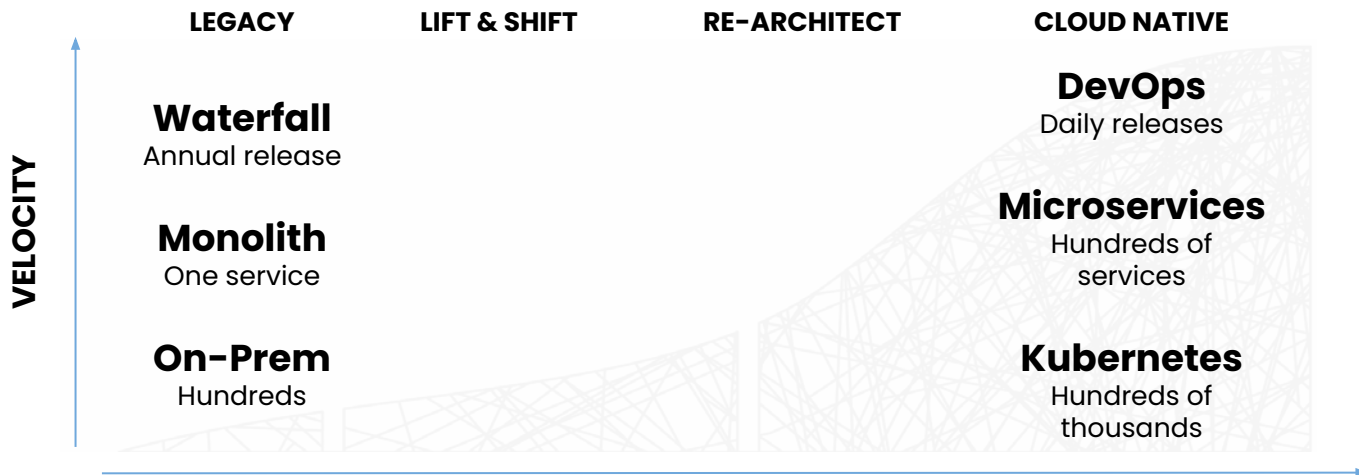


[linkedin.com/in/anammedina](https://www.linkedin.com/in/anammedina)

Our businesses, health,  
and safety rely on applications and  
systems that **will fail**

we've come a long way...

# Complexity



# The Evolution of Chaos Engineering



**NETFLIX**  
F.I.T



**Linked in**  
Simoorg



**amazon**

**UBER**  
uDestroy

**Gremlin**

2010

2012

2014

2016

2018

but...

systems are **complex**  
with **complex failures**



**THIS IS FINE**





**NOW WHAT**



# Continuous Reliability. How?

**AUTOMATION**

**STANDARDIZATION**

**EXPERIMENTATION**



# Ana Margarita Medina

@ana\_m\_medina

- ✨ **Sr. Chaos Engineer** at Gremlin
- ✨ **Developer** since 2007
- ✨ **SRE/Chaos Engineer** since 2016
- ✨ **Keptn Advisory Board** member
- ✨   



keptn



# CLOUD NATIVE COMPUTING FOUNDATION

We are a Cloud Native Computing Foundation Sandbox project.

# Keptn – cloud-native application life-cycle orchestration



## Automating

- Observability, dashboards & alerting
- SLO-driven multi-stage delivery
- Operations & remediation

**declarative, extensible and based on GitOps**

# SLAs and SLOs and SLIs



# SLAs and SLOs and SLIs

**SLAs – Service Level Agreements:** an explicit or implicit contract with your users that includes consequences of meeting (or missing) the SLOs they contain

**SLOs – Service Level Objective:** a target value or range of values for a service level that is measured by an SLI

**SLIs – Service Level Indicator:** a carefully defined quantitative measure of some aspect of the level of service that is provided.

# SLAs and SLOs and SLIs

**Perfect:** 100% of web requests have 0ms latency all the time!

**SLA:** 90% of web requests have latency <500ms for the month... or customer gets money back.

**SLO:** 95% of web requests have latency <500ms over a rolling month.

**SLI:** web requests latency <500ms

# SLOs & SLIs as code

# shift left caring & working on reliability

# “Test Driven Operations” with SLOs


**automate delivery**

**automate SRE  
operations**



63%

Building internal delivery platforms

 **keptn addresses 3 problems raised as adoption challenges in our survey**

95%

time wasted maintaining pipelines

80%

time spent in manual tasks

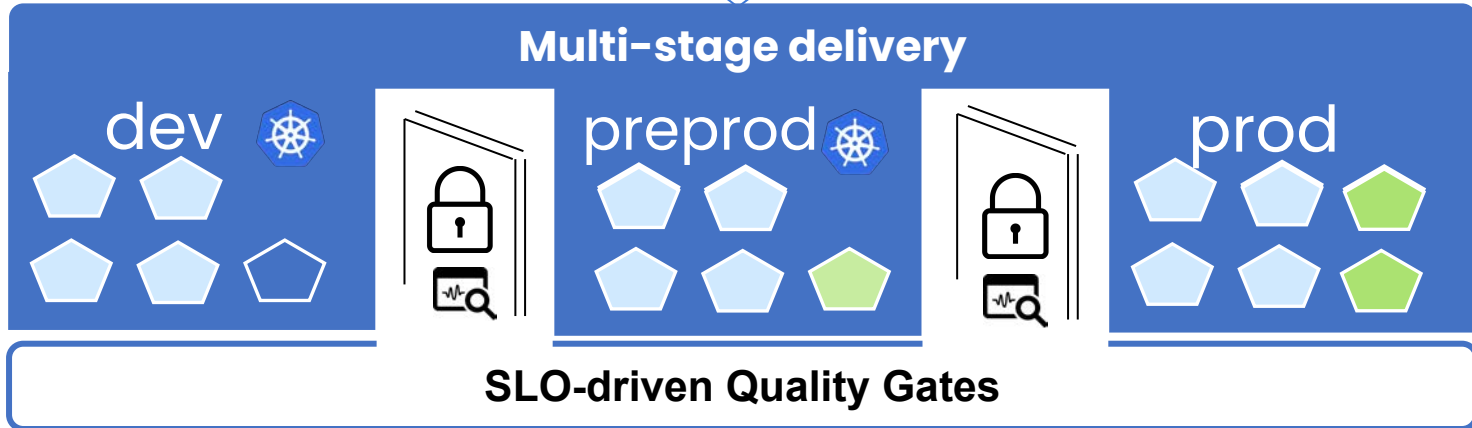
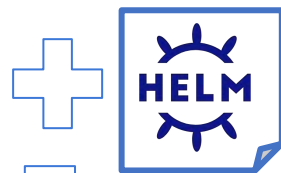
90%

time spent in manual remediation

State of DevOps Report 2020:  
<https://puppet.com/resources/report/2020-state-of-devops-report>

Results are based on dynatrace acsurvey and engagements with Keptn users

Declarative environment  
description &  
service-level objectives

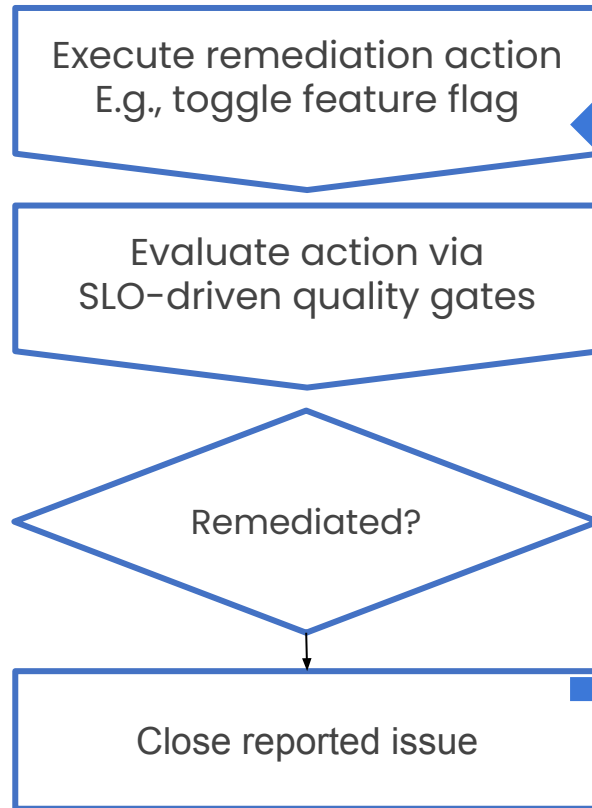






Alert / Problem

Declarative environment description & remediation description



# demo

*a thread*

Keptn Full Tour on Dynatrace

104 mins remaining

- 1 Welcome
- 2 Bring your own Kubernetes cluster
- 3 Install Istio
- 4 Download Keptn CLI
- 5 Install Keptn in your cluster
- 6 Configure Istio and Keptn
- 7 Connect your Keptn CLI to the Keptn installation
- 8 Setup Dynatrace
- 9 Gather Dynatrace tokens
- 10 Deploy Dynatrace OneAgent

# 1. Welcome

In this tutorial you'll get a full tour through Keptn. Before we get started you'll get to know what you will learn while you walk yourself through this tutorial.

## What you'll learn

- How to create a sample project
- How to onboard a first microservice
- How to deploy your first microservice with blue/green deployments
- How to setup quality gates
- How to prevent bad builds of your microservice to reach production
- How to trigger the changes of feature toggles in response to issues detected in a production system
- How to integrate other tools like Slack, MS Team, etc in your Keptn integration

You'll find a time estimate until the end of this tutorial in the right top corner of your screen - this should give you guidance how much time is needed for each step.

In this tutorial, we are going to install Keptn on a Kubernetes cluster, along with Istio for traffic routing and ingress control.

- [Keptn](#) as a control-plane for continuous delivery and automated operations



```
keptn install --endpoint-service-type=ClusterIP --use-  
case=continuous-delivery
```

```
keptn create project sockshop --shipyard=./shipyard.yaml
```

The screenshot shows a web browser window with the URL `http://35.225.198.27.nip.io/bridge/project/sockshop`. The page title is "keptn / Choose project". On the left, there is a navigation menu with "Environment" selected. The main content area is titled "3 Stages" and contains three cards for different stages: "dev", "staging", and "production". Each card displays a progress bar with three icons (a person, a gear, and a shield) and the number "0" next to each, indicating no services are onboarded. Below each progress bar, there is a message: "No service onboarded yet. Follow the instructions to [onboard a service](#)."



```
keptn onboard service carts --project=sockshop --chart=./carts
```

```
keptn onboard service carts-db --project=sockshop --chart=./carts-  
db
```

```
keptn trigger delivery --project=sockshop --service=carts-db --  
image=docker.io/mongo --tag=4.2.2 --sequence=delivery-direct
```

```
keptn trigger delivery --project=sockshop --service=carts --  
image=docker.io/keptnexamples/carts --tag=0.12.1
```

keptn / Choose project

Environment

Services

Sequences

Integration

Filter by

Service

- carts
- carts-db

Stage

- dev
- staging
- production

Sequence

- delivery-direct
- create

Status

- Active
- Failed
- Succeeded

### delivery-direct succeeded

Context: 78e13aac-694c-48ce-8246-4c9fe75909db  
mongo:4.2.2

02:01  dev 02:02  staging 02:02  production

02:02  delivery-direct

02:02  deployment

02:03  release

02:03 shipyard-controller finished with result pass  
Finished release

Last time fetched: today at 02:03:18

**delivery-direct succeeded**  
mongo:4.2.2 in **dev** - **staging** - prod

Started today at 02:01 and took 1 minutes 37 seconds  
Last task: delivery-direct

**create succeeded**  
carts-db

Started today at 02:00 and took 8 seconds  
Last task: create

**create succeeded**  
carts

Started today at 01:59 and took 7 seconds  
Last task: create

**create succeeded**

Started today at 01:58 and took 9 seconds  
Last task: create

keptn / Choose project v

Environment: Filter by

Services:  carts,  carts-db

Sequences:  dev,  staging,  production

Integration:  delivery,  monitoring,  delivery-direct,  create

Status:  Active,  Failed,  Succeeded

**delivery started**  
carts:0.12.2 in dev 0 staging 0  
Started today at 02:18 and running for 14 minutes  
24 seconds  
Recent task: rollback

**monitoring succeeded**  
Started today at 02:16 and took 3 seconds  
Last task: configure-monitoring

**delivery succeeded**  
carts:0.12.1 in dev 0 staging 0 produ  
Started today at 02:03 and took 8 minutes 47 seconds  
Last task: delivery

**delivery-direct succeeded**  
mongo4.2.2 in dev 0 staging 0 produ  
Started today at 02:01 and took 1 minutes 37 seconds  
Last task: delivery-direct

**create succeeded**  
carts-db  
Started today at 02:00 and took 8 seconds  
Last task: create

**create succeeded**  
carts  
Started today at 01:59 and took 7 seconds  
Last task: create

keptn provides an automatic version check to be notified about new versions and security patches for Keptn. It is recommended to have this check enabled. This setting can be changed in the user menu on the top right.  
[Accept \(recommended\)](#) [Decline](#)

Pod name: carts-69c578667c-7794h  
Container image: docker.io/keptnexamples/carts:0.12.2  
Deployment name: carts  
keptn project: carts  
keptn stage: sockshop-dev  
keptn service: carts  
Version: v2  
Delay in ms: 1000  
Promotion rate: 0

Pod name: carts-57dbd57446-qd8mz  
Container image: docker.io/keptnexamples/carts:0.12.2  
Deployment name: carts  
keptn project: carts  
keptn stage: sockshop-staging  
keptn service: carts  
Version: v2  
Delay in ms: 1000  
Promotion rate: 0

Pod name: carts-primary-59968bb6d5-614d7  
Container image: docker.io/keptnexamples/carts:0.12.1  
Deployment name: carts  
keptn project: carts  
keptn stage: sockshop-production  
keptn service: carts  
Version: v1  
Delay in ms: 0  
Promotion rate: 0



← → ↻ Not Secure | http://35.225.198.27.nip.io/bridge/project/sockshop

keptn / Choose project ▾

Environment  
Services  
Sequences  
Integration

### 3 Stages

#### dev

! 0 🛠 0 🛡 0

carts:0.12.2	📦 delivery	0
mongo:4.2.2	📦 delivery-dire	-

#### staging

! 0 🛠 1 🛡 0

carts:0.12.2	📦 delivery	0
mongo:4.2.2	📦 delivery-dire	-

#### production

! 0 🛠 0 🛡 0

carts:0.12.1	📦 delivery	-
mongo:4.2.2	📦 delivery-dire	-

Show SLO Ignore for comparison

#### Evaluation of test on staging

0 < 75 Result: fail

**Evaluation timeframe:** 2021-05-20 02:23 - 2021-05-20 02:32 (8 minutes 44 seconds)

Compared with last evaluation, 2 results are not showing up in the Heatmap, because they have been either invalidated or are older results.

SLI breakdown

Name	Value	pass Criteria	warning Criteria	Result	Score
response_time_p95	1052	<=+10% and <600	<<800	failed	0

[Show SLO](#)[Ignore for comparison](#)

## 📅 Evaluation of test on staging

0 < 75 Result: fail

**Evaluation timeframe:** 2021-05-20 02:23 - 2021-05-20 02:32 (8 minutes 44 seconds)

Compared with last evaluation. 2 results are not showing up in the Heatmap, because they have been either invalidated or are older results.

### SLI breakdown

Name ↕	Value	pass Criteria	warning Criteria	Result	Score ▲
response_time_p95	1052	<=+10% and <600	<=800	failed	0

keptn / Choose project ▾

Environment

Services

Sequences

Integration

Filter by

Service

- carts
- carts-db

Stage

- dev
- staging
- production

Sequence

- delivery
- monitoring
- delivery-direct
- create

Status

- Active
- Failed
- Succeeded

**delivery succeeded**

Context: f2cbd217-03c1-42da-9c94-8d7982c13f36  
carts:0.12.3

02:37                                      02:40                                      02:47

02:47 **delivery**

02:47 **deployment**  
Labels: DiCreds: dynatrace

02:48 **% release**  
Labels: DiCreds: dynatrace

02:50 shipyard-controller finished with result pass  
Finished release

Last time fetched: today at 02:52:03

Pod name: **carts-6d4ccbd4f9-4byk4**  
 Container image: **docker.io/keptnexamples/carts:0.12.3**  
 Deployment name:  
 keptn project: **carts**  
 keptn stage: **sockshop-dev**  
 keptn service: **carts**  
 Version: **v3**  
 Delay in ms: **0**  
 Promotion rate: **0**

cars.sockshop-staging-35.21

Not Secure | http://cars.sockshop-...

Pod name: **carts-6ff675db5d-27f8z**  
 Container image: **docker.io/keptnexamples/carts:0.12.3**  
 Deployment name:  
 keptn project: **carts**  
 keptn stage: **sockshop staging**  
 keptn service: **carts**  
 Version: **v3**  
 Delay in ms: **0**  
 Promotion rate: **0**

cars.sockshop-production.31

Not Secure | http://cars.sockshop-...

Pod name: **carts-6776ccb7df-z58p6**  
 Container image: **docker.io/keptnexamples/carts:0.12.3**  
 Deployment name:  
 keptn project: **carts**  
 keptn stage: **sockshop-production**  
 keptn service: **carts**  
 Version: **v3**  
 Delay in ms: **0**  
 Promotion rate: **0**

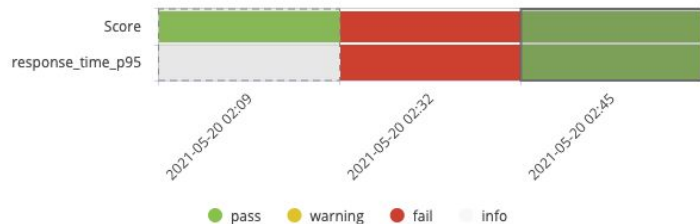
Stage: staging

### evaluation

2021-05-20 02:45

Labels: DtCreds: dynatrace

Heatmap Chart



● pass ● warning ● fail ● info

Show SLO Ignore for comparison

### Evaluation of test on staging

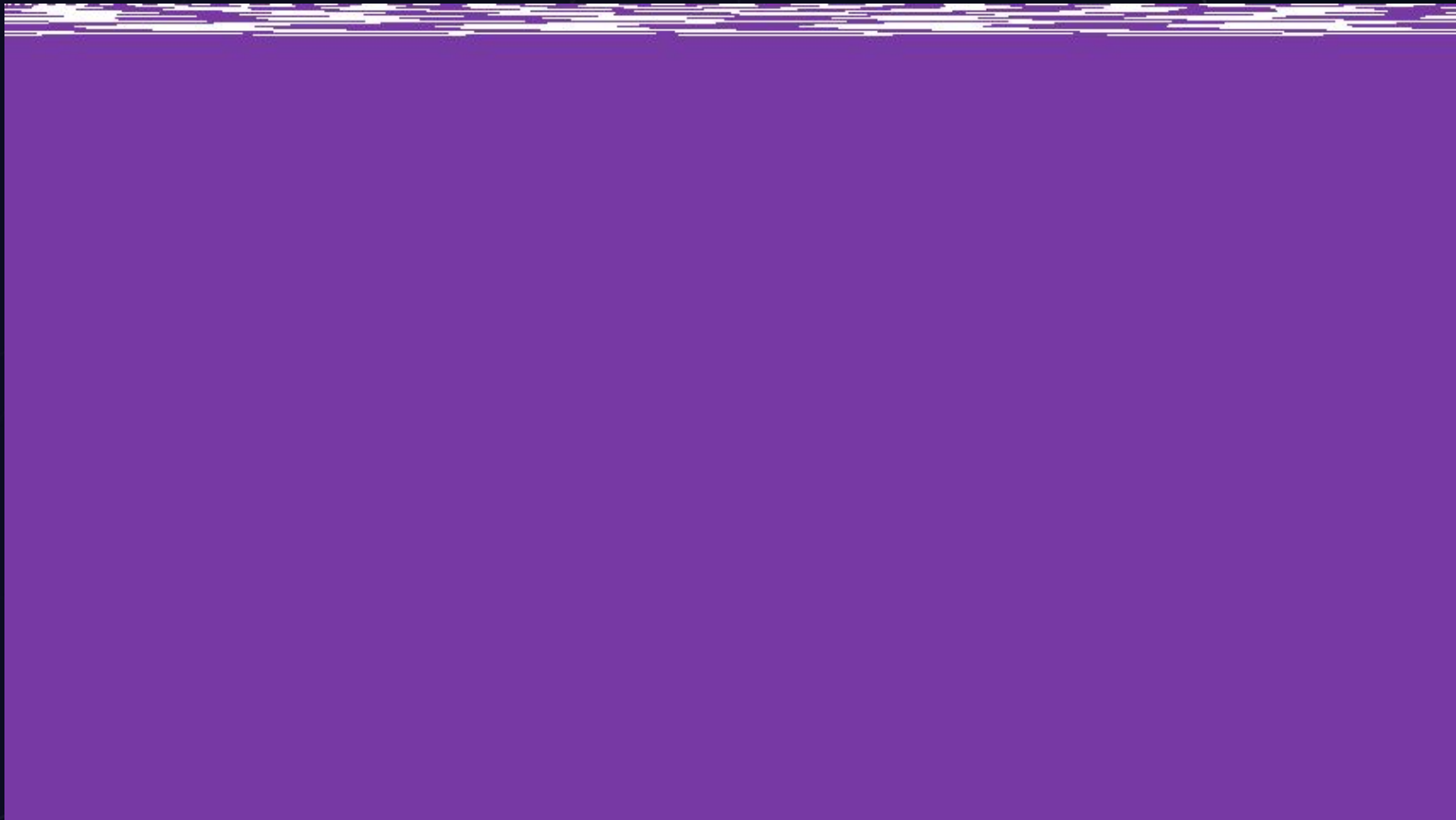
90 <= 100 Result: pass

Evaluation timeframe: 2021-05-20 02:41 - 2021-05-20 02:43 (1 minutes 56 seconds)

Compared with last evaluation.

#### SLI breakdown

Name	Value	pass Criteria	warning Criteria	Result	Score
------	-------	---------------	------------------	--------	-------



# SLOs & Keptn & Chaos Engineering

# Chaos Engineering on Keptn

**You can run Chaos Engineering experiments:**

- on every stage
- on a chaos stage
- alongside performance testing

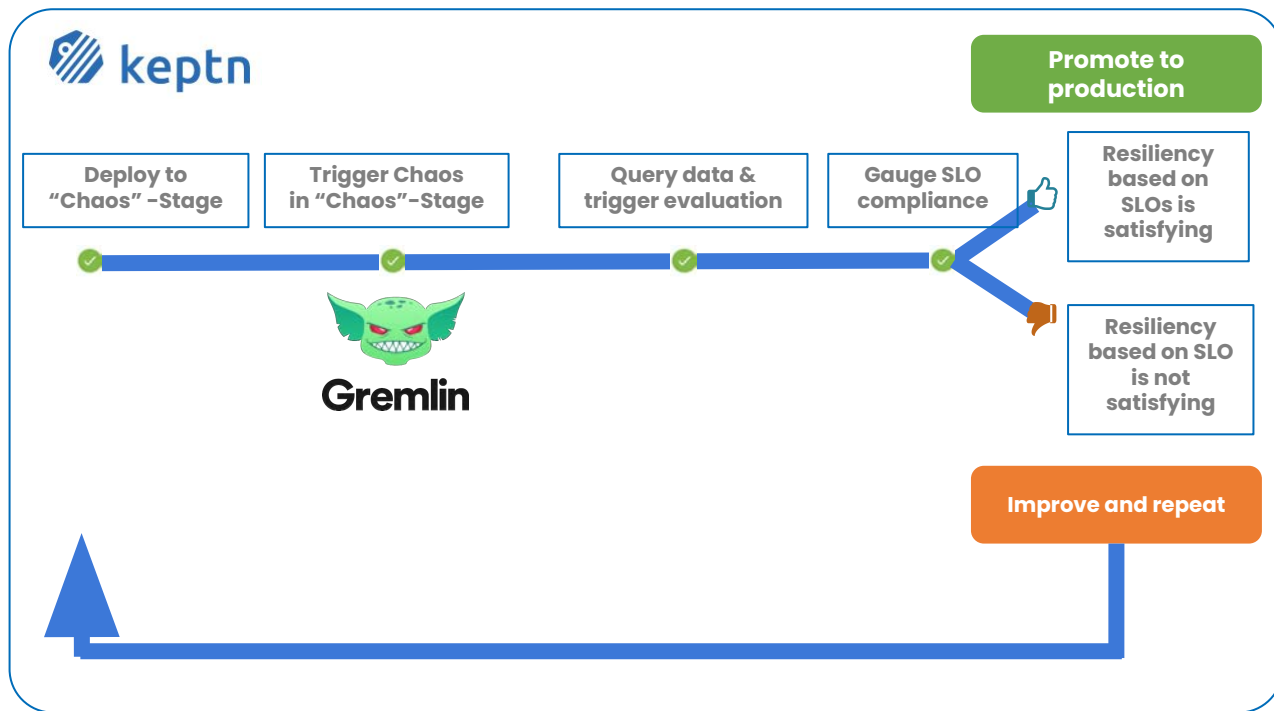
# Chaos Engineering on Keptn

## Are SLOs still met?

















- SLOs is met
  - Promote to Production
- SLOs is NOT met
  - **Did we identify a weakness?**
  - Improve and repeat



# Chaos Engineering on Keptn



# Integrations with:

Delivery					
Test					
Observability					
Collaboration					

... and growing!

# Join the Keptn community

- **Visit** [keptn.sh](https://keptn.sh)
- **Follow** [twitter.com/KeptnProject](https://twitter.com/KeptnProject)
- **Give a star** [github.com/keptn/keptn](https://github.com/keptn/keptn)
- **Get your hands dirty** [tutorials.keptn.sh](https://tutorials.keptn.sh)
- **Continue Learning** [youtube.com/c/keptn/](https://youtube.com/c/keptn/)



keptn

**You can't build**  
**Reliability**  
**over night**

**You can't buy**  
**Reliability**  
**over night**

# Build **Reliability** by establishing a process, automating it, and continuously validate the processes

*(Experiments, SLOs, Failovers,  
Runbooks)*

**Reliability**  
**is not an**  
**accident**

Are you ready to become  
a Gremlin-certified Chaos  
Engineering Practitioner?

---

[gremlin.com/certification](https://gremlin.com/certification)





The screenshot shows a web browser window displaying the Gremlin website. The browser's address bar shows 'gremlin.com/certification/'. The website's navigation bar includes links for 'Product', 'Pricing', 'Resources', 'Community', 'Company', and 'Login', along with a 'GET STARTED' button. The main content area features the 'Gremlin Foundations Series' heading, followed by a description: 'Learn the foundations of Chaos Engineering and testing for Reliability with Gremlin.' To the right, there are two certification levels: 'PRACTITIONER' and 'PROFESSIONAL', each with a brief description and a circular icon. Below this, a large image shows a sample certificate for Billy Peltzer, dated September 12, 2022, certifying him as a 'CERTIFIED CHAOS ENGINEERING PRACTITIONER'. To the right of the certificate, the text reads 'Show the world your Chaos Engineering skills.' and 'Display in your office'. Below that, it says 'Share on social media' and 'Backed by experts'. The bottom of the page contains a small paragraph stating that the certificate program was designed by Chaos Engineering leaders and pioneers with over 150+ years of combined experience.

## Gremlin Foundations Series

Learn the foundations of Chaos Engineering and testing for Reliability with Gremlin.

### PRACTITIONER

Start building Chaos Engineering expertise, increase your professional visibility, and advance your career with a Gremlin Chaos Engineering Practitioner certification.

### PROFESSIONAL

Build on what you learned from the GCEEP program with a course that focuses on more advanced Chaos Engineering best practices and more expert ways to use Gremlin.

## Show the world your Chaos Engineering skills.

Build your reputation as one of the world's best Chaos Engineering practitioners with an easily shareable certificate that's backed by the industry's leaders and pioneers.

### Display in your office

After you pass the Gremlin Chaos Engineering Practitioner Exam, we'll ship the first 50 certified practitioners a copy of your certificate to proudly display in your office.

### Share on social media

After you pass the Gremlin Chaos Engineering Practitioner Exam, we'll ship the first 50 certified practitioners a copy of your certificate to proudly display in your office.

### Backed by experts

The certificate program was designed by Chaos Engineering leaders and pioneers with over 150+ years of combined experience.

**Thank you!** 🙌

**Questions?**

ana@gremlin.com  
@ana\_m\_medina

