INTRODUCTION TO
CHAOS ENGINEERING

Ana Medina
Chaos Engineer at Gremlin
@ana_m_m_medina
I work at Gremlin as a Chaos Engineer.

I was a Software Engineer at Uber in SRE doing Chaos Engineering and Cloud Computing.

I'm from Costa Rica and Nicaragua and currently living in San Francisco, CA.

College dropout.
Self taught engineer
Anything that can go wrong....
will go wrong.

- Murphy’s Law
Chaos Engineer /ˈkɑː,æs/,ˌenjəˈnɪr/
noun
1. a person helping companies avoid outages by running proactive chaos engineering experiments.
As an SRE, we want to have confidence that our systems are resilient or even antifragile.

Confidence
1. full trust; belief in the powers, trustworthiness, or reliability of a person or thing:

Resilient
1. springing back; rebounding
2. Returning to the original form or position after being bent, compressed, or stretched.
“Some things benefit from shocks; they thrive and grow when exposed to volatility, randomness, disorder, and stressors and love adventure, risk, and uncertainty. Yet, in spite of the ubiquity of the phenomenon, there is no word for the exact opposite of fragile. Let us call it antifragile. Antifragility is beyond resilience or robustness. The resilient resists shocks and stays the same; the antifragile gets better.”

-Nassim Nicholas Taleb
Antifragile: Things That Gain from Disorder
How to make our systems antifragile?

BREAK THINGS ON PURPOSE.
CHAOS ENGINEERING
We test proactively, instead of waiting for an outage.

-Kolton Andrus
CEO, Gremlin
Thoughtful, planned experiments designed to reveal the weakness in our systems.
WHAT CHAOS ENGINEERING IS

- Controlled and planned chaos engineering experiments
- Preparing for unpredictable failure
- Preparing engineers for failure
- Preparing for game days
- A way to improve SLA
  - fortify systems
  - build and move fast
  - build confidence in systems
  - reveal weak points in your systems
  - build assurance that you can still serve your customers
WHAT CHAOS ENGINEERING IS NOT

- Random chaos engineering experiments
- Unsupervised chaos engineering experiments
- Unmonitored chaos engineering experiments
- Unexpected chaos engineering experiments
- Breaking production by accident
- Creating Outages
WHY DO CHAOS ENGINEERING?

- Microservice Architecture is tricky
- Our systems are scaling fast
- Services will fail
- Dependencies on other companies will fail
- Prepares for real world scenarios
- Reduce amount of outages, reduce down time, lose less money
WHY DO CHAOS ENGINEERING?

- Human Factor: Train on-call engineers

  **Human Factor**
  1. It accounts for the user's capabilities and limitations in seeking to ensure that tasks, functions, information, and the environment suit that user.

Train engineers in a healthy way:
  a. maintaining quality of life
  b. their mental health
  c. avoid burnout / turnover
If you haven't tried it, assume it's broken.

-Unknown
WHAT COMPANIES ARE DOING IT?

- Netflix
- Amazon
- Dropbox
- Uber
- Slack
- Twilio
- And more!
WHERE CAN YOU IMPLEMENT THIS?

- Application
- API
- Caching
- Database
- Hardware
- Cloud Infrastructure / Bare metal
WHAT SERVICES/TEAMS SHOULD DO THIS?

● Critical services - (Tier 0 services)
  ○ Top 5 critical services is a good way to get started
● Critical data storage
@ YOUR COMPANY /
WHAT CAN YOU DO?

- Unpredictable circumstances
- Large traffic spikes
- Race conditions
- Datacenter failure
- Time travel - system clocks to be out of sync
- Network errors
- CPU overloads
@ YOUR COMPANY / HOW?
Minimize the blast radius.

Form a hypothesis.

Run an experiment.

Abort Conditions

Failure

Success

Find and fix issues.

Scale up and repeat.
HOW ARE PEOPLE DOING IT?

ChaosMonkey
KubeMonkey
Gremlin
WHERE TO LEARN MORE

Principles of Chaos
Chaos Engineering Book
Awesome Chaos Engineering GitHub Repo
Chaos Engineering Slack  gremlin.com/slack
Gremlin Community
Netflix - Chaos Kong
Chaos Engineering Meetups
Learn More about SEV
Learn More about GameDays

Get Involved: come to ChaosConf in San Francisco, CA - chaosconf.io
ARE YOU READY?
THINGS NEEDED TO DO CHAOS ENGINEERING:

- High Severity Incident Management - “SEVs”
- Useful monitoring
- Alerting and paging
- Clear instructions on how to roll back an experiment
- Measure the impact of downtime
THANKS!

#BreakThingsOnPurpose

@ana_m_medina
ana@gremlin.com

www.gremlin.com

@gremlininc