DW6

Microservices & Cloud
Wednesday, November 7th, 2018 1:30 PM

Creating Chaos: Engineering for the Unexpected

Presented by:

Shahzad Zafar

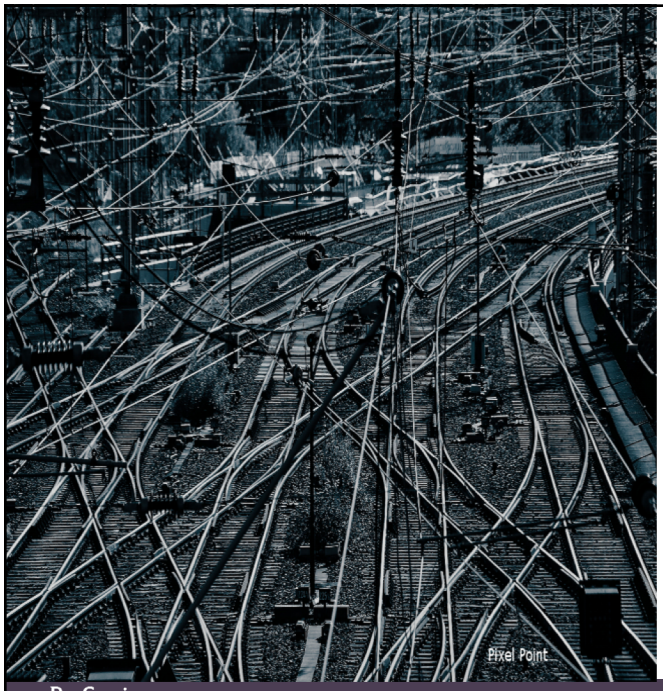
RxSavings

Brought to you by:

350 Corporate Way, Suite 400, Orange Park, FL 32073
888-268-8770 · 904-278-0524 - info@techwell.com - http://www.starwest.techwell.com/

Shahzad Zafar

Shahzad Zafar is the Vice President of Engineering at Rx Savings Solutions. Before joining Rx Savings Solutions in 2018, he worked at Cerner for 13 years, where he led the Cloud Platform development business unit while being an agile coach as well.. Shahzad has a degree in computer engineering from the University of Michigan, Ann Arbor, and received his master's in business administration from the University of Kansas. Shahzad is also a board member for AgilehoodKC and speaks regularly at Meetups and conferences such as LeanAgileKC, KCPMI PDD, Agile Midwest St. Louis, and Kansas City Developers Conference. He also teaches classes around Information Technology in the University of Kansas Business School's Graduate program.



Creating Chaos…

Engineering for the Unexpected!

Shahzad Zafar
Vice President of Engineering
Rx Savings Solutions

Creating Chaos … Engineering!

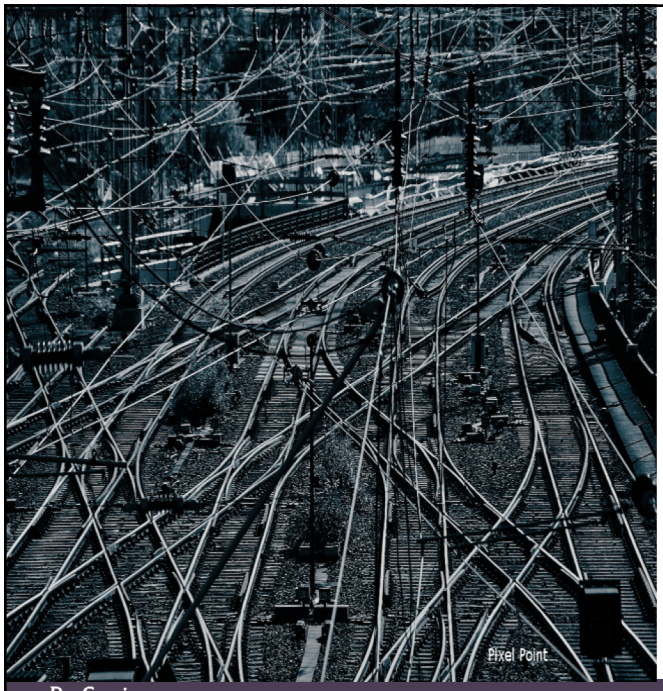
Go to **www.menti.com** and use the code **51 76 0**

What does Chaos Engineering mean to you?

Mentimeter

Slide is not active Activate

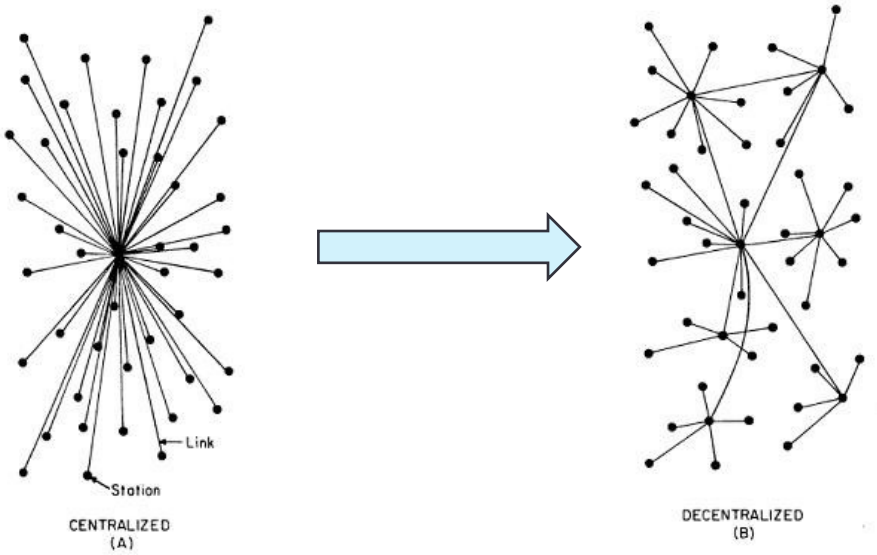
21



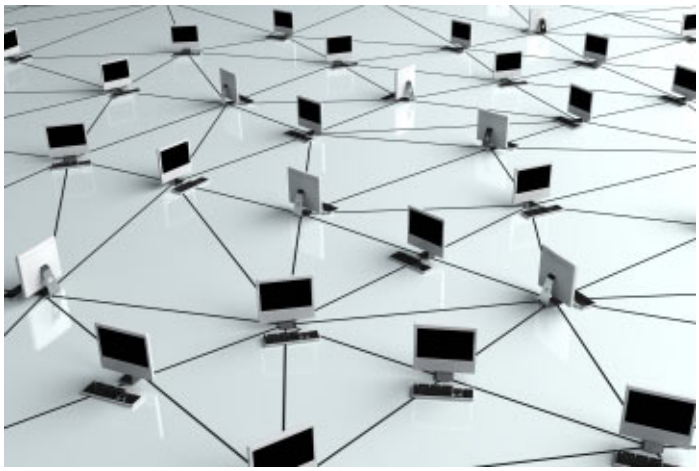
Creating Chaos … Engineering!

Shahzad Zafar
Vice President of Engineering
Rx Savings Solutions

Why This Topic?



Why This Topic?



"A distributed system is one in which the failure of a computer you didn't even know existed can render your own computer unusable" - Leslie Lamport

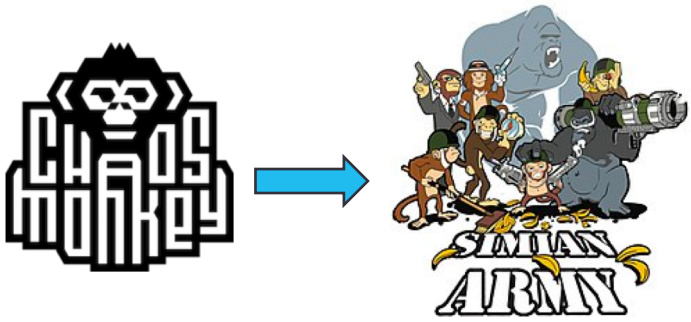
What is Chaos Engineering?

What is Chaos Engineering?

What is Chaos Engineering?

- Requires
  - Having a hypothesis
  - Identifying control conditions
  - Uses real-world events
  - Limiting the scope or blast radius
  - Make it as real as possible
    - Ideally running it in Prod

Chaos Monkey vs. Chaos Engineering



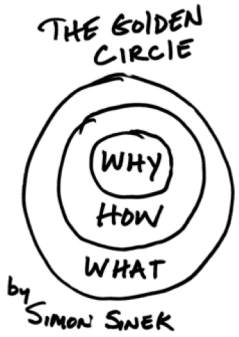
Chaos Engineering

NETFLIX

Chaos Monkey
Chaos Gorilla
Chaos Kong
Janitor Monkey
Doctor Monkey
Compliance Monkey
Latency Monkey
Security Monkey

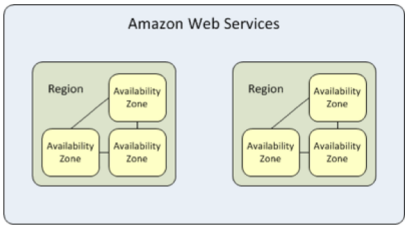
Principles of Chaos Engineering (aka running the experiments)

- #1 Have a Good Hypothesis
  - Start with the Why?
  - Like any experiment, know what is the expected behavior
- #2 Use Real-World Events
  - Use frequent and/or high impact scenarios
  - Review incidents and use them refine scenarios
- #3 Continuous Experimentation
  - Automate the process of running experiments
  - Tools to both orchestrate and analyze experiments



Principles of Chaos Engineering (aka running the experiments)

- #4 Use Business Metrics
  - Start with steady state system metrics such as throughput, error rates etc. **(outputs)**
  - Move quickly to using business metrics such as value added, functionality usage **(outcomes)**
- #5 Limiting Blast Radius
  - Goal is not to experiment against the whole system
  - Scale the experiment up and stop when it starts impacting business metrics



- #6 Run Experiments in Production
  - Most realistic setup is in Production
  - Use principles #4 and #5 to avoid impacting users

Where to Start?

- Start with Known Weakest Link
  - Helps in building practice and muscle memory
  - Work your way backwards to find the unknowns

- Monitoring
  - First few times could be manual monitoring
    - As long as monitoring steps are accounted for in the hypothesis
  - Quickly automate, so you can focus on anomalies during an experiment

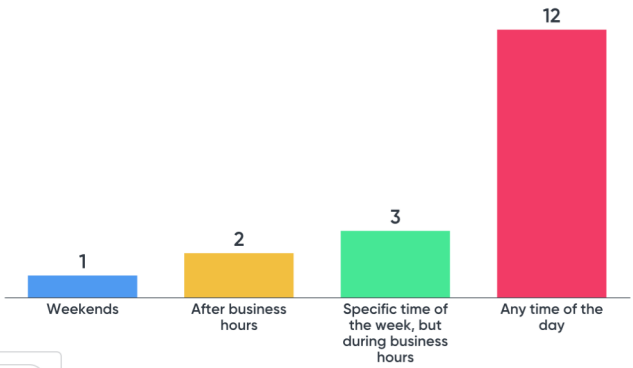
- Being Inclusive
  - Humans are part of the system … test them
  - Find your Brent (from the Phoenix Project)

Risk Tolerance

Go to **www.menti.com** and use the code **51 76 0**

When do you usually make production changes?

Mentimeter



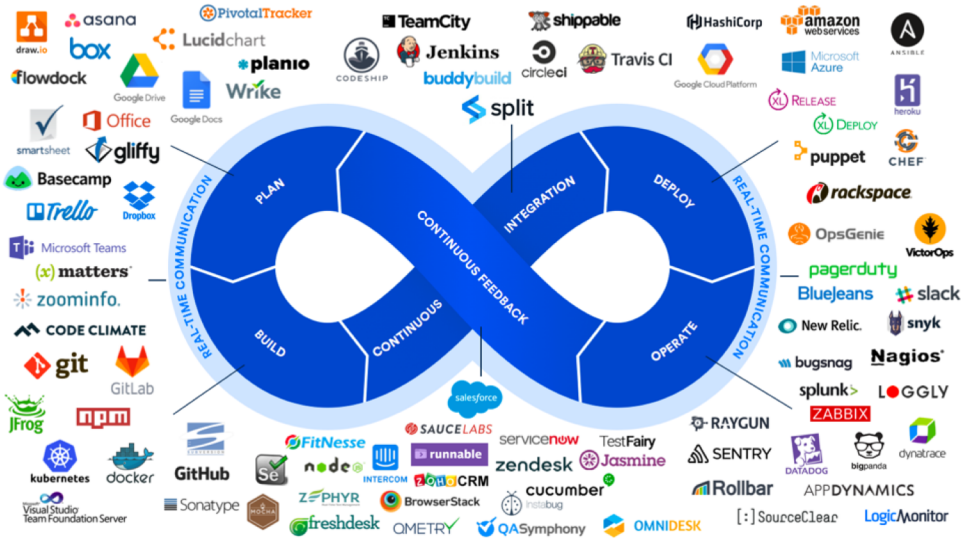
Slide is not active Activate

18

Where to Start?

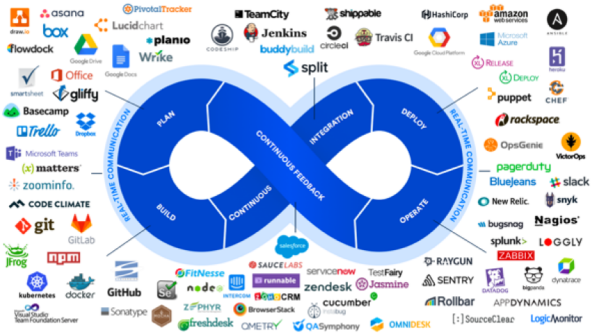
- Organizational Risk Tolerance
  - Starting with planned, announced events
  - Run enough experiments to improve tolerance
  - High risk times is when to run the experiments
    - Work to be done in “off” hours should not be acceptable
    - Build our system to be resilient to any change at any time
  - Goal: build resilient products
    - By running unannounced experiments, all the time
- Understanding the process of creating hypothesis

DevOps & Chaos Engineering



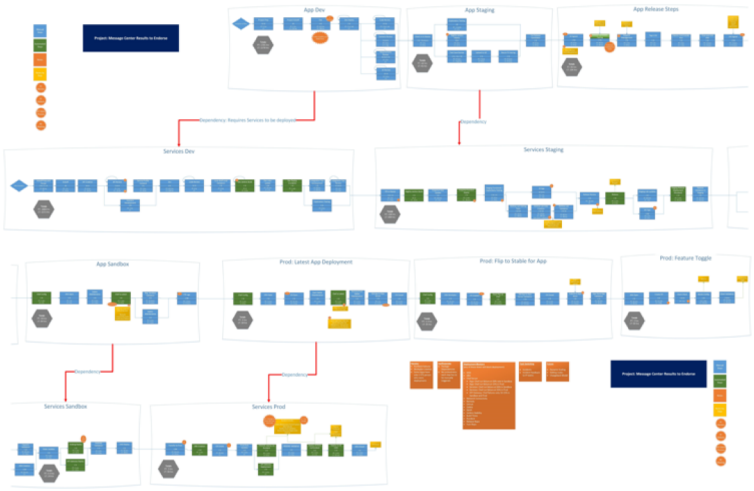
DevOps & Chaos Engineering

- Given the ever increasing toolset
  - Need vertical alignment from inception to delivery
  - DevOps mindset and behaviors are needed truly chaos test your system
  - System monitoring and operations need to be built-in as features from the beginning
  - 1 in 2^n chance of success
    - Where n is the number of dependencies
    - Troy Magennis – Agile2018 Keynote



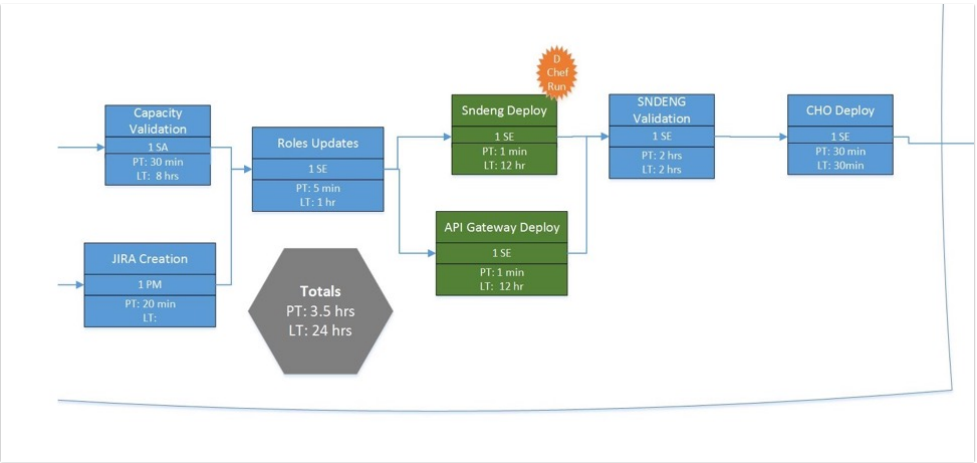
DevOps & Chaos Engineering

- Value Stream Mapping
  - Map out the entire system to find bottlenecks and weak spots



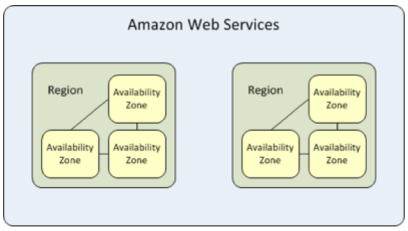
DevOps & Chaos Engineering

- Value Stream Mapping
  - Map out the entire system to find bottlenecks and weak spots



Real Experiments

- Test failure of a load balancer or service
  - Identify resiliency at an individual component level
- Fault testing for an Availability Zone or Region
  - Identify failover resiliency
- Test failure of an entire rack
  - Identify resiliency when several components fails



Real Experiments

- Power Loss vs. Server Shutdown

- In our first experiment, hypothesis was it would have the same result
- Pulling the power out revealed some other dependencies that did not show up when just shutting down a server

Scaling Beyond a Team

- Moving from ”The Shadows” to Invested
  - Pilot is small and might not need approvals, beyond team buy-in
  - Getting investment helps in broader buy-in and support to build tooling around it
- Creating an Automation Tool, which can
  - Do canary analysis
  - Have default monitoring and controls
- Get to a point where running an experiment needs to be
  - Routine
  - Not time consuming

Conclusions

- Start small, grow from there
- Spend time writing your hypothesis
- Automate and build-in needed capabilities
- Recognize risk tolerance
  - And get comfortable running experiments during ‘high risk’ times
- Run experiments all the time

And to ensure system resiliency…

Create Chaos!

References

- Chaos Engineering
  - Building Confidence in System Behavior through Experiments
  - https://www.oreilly.com/webops-perf/free/chaos-engineering.csp
- Canary Analyze All The Things
  - https://www.infoq.com/presentations/canary-analysis-deployment-pattern
- The Phoenix Project
  - https://www.amazon.com/Phoenix-Project-DevOps-Helping-Business/dp/0988262592
- A comprehensive guide by Gremlin
  - https://www.gremlin.com/chaos-monkey/
- Performing Chaos at Netflix Scale
  - https://www.youtube.com/watch?v=LaKGx0dAUlo

Thank You!

Shahzad Zafar
@m_shahzad_z

Rx Savings Solutions