Align the Introduction of Chaos with Organized Experiments

Optimize engineering focus on the introduction of chaos as planned experiments.

Minimize the opportunity for chaos to become a scape goat for mysterious issues.

Prepare for the Experiment

Describe the scenario, what is expected to occur, how it will be measured, who is needed.

Identify prerequisites that are needed to be completed
(ex. improved telemetry on connection refresh of data store)

Plan to be Surprised

We generally always learned something new about the larger system and the effects of compounding failures.

Capture what was surprising (actual results vs. what was the expected results) in an open and searchable repository.

Plan added time to digest the surprises.

Observability is Critical

You need easy access to essential telemetry data of all the parts of the system.

You want to be able to ask different and new questions of your system without having to change the system.

When you discover a gap in visibility, focus on how to make it easy to rebuild your system with the improvement through low coordination.

Utilize a Dedicated Space

Have a common space (physical/virtual) where everyone attends during the experiment.

You want to optimize communication when assessing the experiment. Schedule adequate amount of time for multiple iterations (ex. whole afternoon).

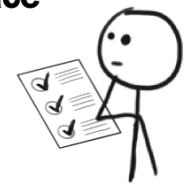
Prepares Your Team

Your entire team may not be able to participate, but they should be able to learn from the findings.

Experiments help you practice how you look into the system, where signals normally arise, and identifies gaps on essential telemetry for broader insight.

Understand and Embrace needed Compliance

Production systems will bear more compliance and controls.



Much of this is around risk, so focus on the introduction through low risk scenarios (ex. non-live systems being built).

Cross Functional Involvement

Helps share knowledge on how different layers of a system are viewed during the experiment.

Diverse perspectives can accelerate and improve group learning.

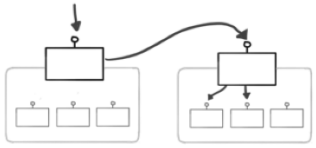
Getting Started with Chaos Experiments

Carl Chesser
@che55er | che55er.io

Traffic Management Patterns

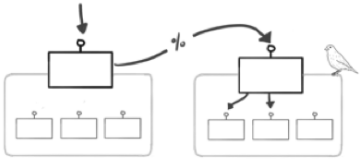
Chaining Traffic

Supports an API gateway to simply call another gateway, versus the backing set of services.



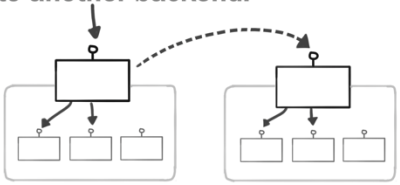
Canary Traffic

Supports gradually transitioning a subset of traffic to a different target by leveraging chaining.



Shadowing Traffic

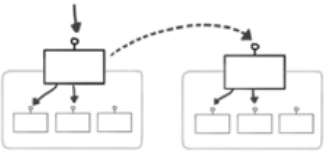
Replays a percentage of traffic to another backend.

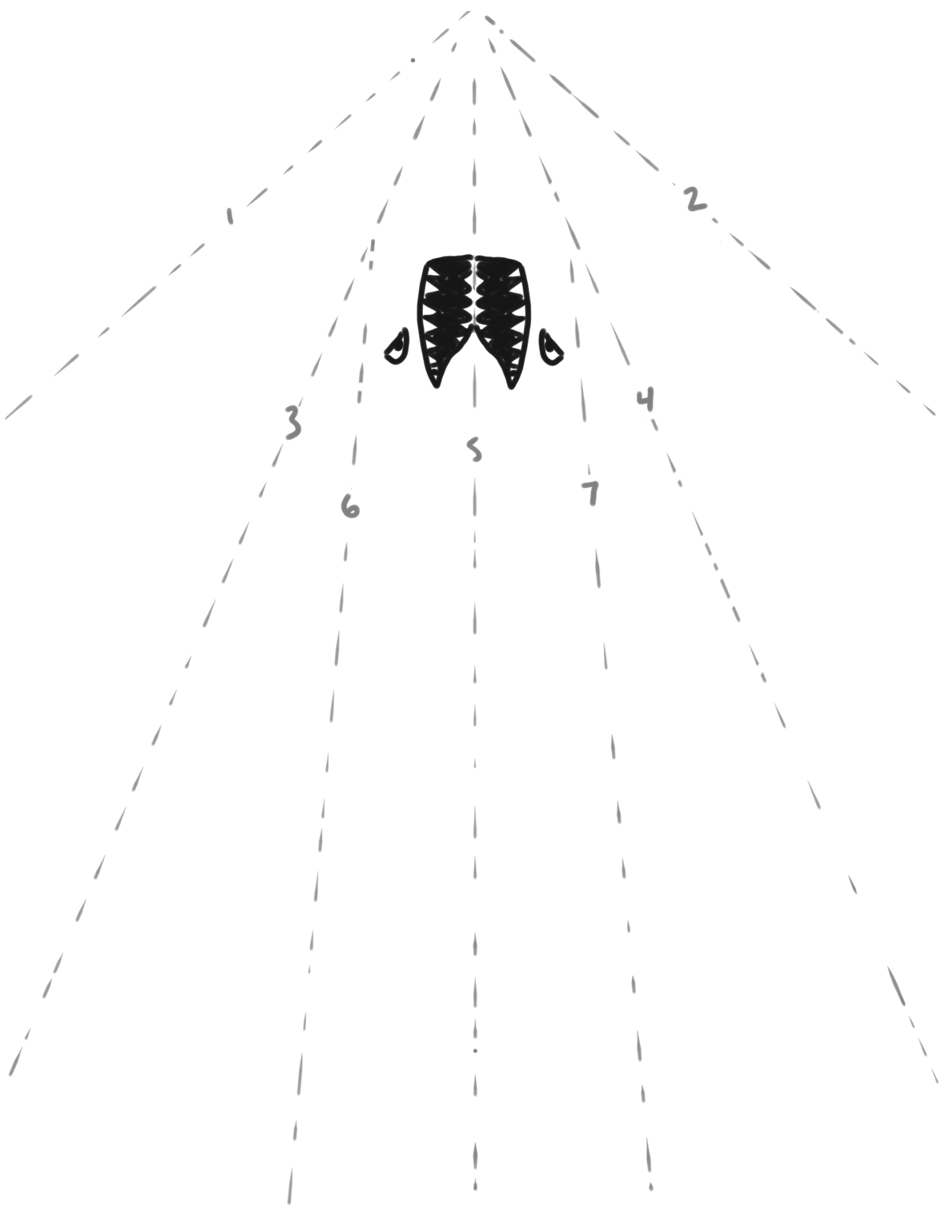


Background replay of safe requests. (read-only, HTTP GET)

Shadow Allows Early Testing

Rather than imposing a canary early with experiments, where a small percentage of failure still introduces undesirable risk, look to leverage a shadow of traffic.




1
2
3
4
5
6
7