

# Chaos Engineering Saved Your Netflix

## Extreme stress testing of online platforms has become its own science

[Michael Dumiak](#)

03 Mar 2021

5 min read



Photo: Pavlo Gonchar/Sipa/AP

To hear Greg Orzell tell it, the original [Chaos Monkey](#) tool was simple: It randomly picked a virtual machine hosted somewhere on [Netflix's](#) cloud and sent it a "Terminate" command. Unplugged it. Then the [Netflix](#) team would have to figure out what to do.

That was a decade ago now, when Netflix moved its systems to the cloud and subsequently navigated itself around a major U.S. East

Coast service outage caused by its new partner, [Amazon Web Services](#) (AWS).

Orzell is currently a principal software engineer at [GitHub](#) and lives in Mainz, Germany. As he recently recalled the early days of Chaos Monkey, Germany got ready for another long round of COVID-related pandemic lockdowns and deathly fear. Chaos itself raced outside.

But while the coronavirus wrenched daily life upside-down and inside out, a practice called chaos engineering, applied in computer networks, might have helped many parts of the networked world limp through their coronavirus-compromised circumstances.

Chaos engineering is a kind of high-octane active analysis, stress testing taken to extremes. It is an emerging approach to evaluating distributed networks, running experiments against a system while it's in active use. Companies do this to build confidence in their operation's ability to withstand turbulent conditions.

Orzell and his Netflix colleagues built [Chaos Monkey](#) as a Java-based tool from the [AWS software development kit](#). The tool acted almost like a number generator. But when Chaos Monkey told a virtual machine to terminate, it was no simulation. The team wanted systems that could tolerate host servers and pieces of application services going down. "It was a lot easier to make that real by saying, 'No, no, no, it's going to happen,' " Orzell says. "We promise you it will happen twice in the next month, because we are going to make it happen.' "

## Spawning Chaos

Netflix originated "chaos engineering." Here's how they do it.

A Netflix engineer uses the company's Chaos Automation Platform (ChAP) (1) to connect with its continuous-delivery system (2). The system reaches Netflix's external cloud-service provider (3), which, per ChAP's orders, will slightly modify operations on a number of test subjects to determine where the stress points in the system's "control plane" (4) are. When subscribers cross the service's virtual front door (5) and select a video to watch, a few are quietly delivered the altered service as part of their viewing experience. (One example Netflix describes is intentionally altering the bookmarking service for its test users: If they exit from Netflix in the midst of watching a video, the service might lose its ability to remember the users' stopping point.) ChAP then instructs its real-time monitoring system (6) to observe the test users' experience to ensure that ChAP's small errors are compensated for and don't result in cascading failures or crashes. Source: A. Basiri et al., "Automating Chaos Experiments in Production," proceedings of the 41st International Conference on Software Engineering, 2019

In controlled and small but still significant ways, chaos engineers see if systems work by breaking them—on purpose, on a regular basis. Then they try to learn from it. The results show if a system works as expected, but they also build awareness that even in an engineering organization, things fail. All the time.

As practiced today, chaos engineering is more refined and ambitious still. [Subsequent tools](#) could intentionally slow things down to a crawl, send network traffic into black holes, and turn off network ports. (One related app called Chaos Kong could scale back company servers inside an entire geographic region. The system would then need to be resilient enough to compensate.) Concurrently, engineers also

developed guardrails and safety practices to contain the blast radius. And the discipline took root.

At Netflix, chaos engineering has evolved into a platform called the Chaos Automated Platform, or ChAP, which is used to run [specialized experiments](#). (See "Spawning Chaos," above.) Nora Jones, a software engineer, founder and chief executive of a startup called [Jeli](#), says teams need to understand when and where to experiment. She helped implement ChAP while still at Netflix. "Creating chaos in a random part of the system is not going to be that useful for you," she says. "There needs to be some sort of reasoning behind it."

Of course, the novel coronavirus has added entirely new kinds of chaos to network traffic. Traffic fluctuations during the pandemic did not all go in one direction either, says AWS principal solutions architect Constantin Gonzalez. Travel services like the German charter giant Touristik Union International (TUI), for instance, drastically pulled in its sails as traffic ground to a halt. But the point in building resilient networks is to make them elastic, he says.

Chaos engineering is geared for this. As an engineering mind-set, it alludes to [Murphy's Law](#), developed during moonshot-era rocket science: If something can go wrong, it will go wrong.

It's tough to say that the practice kept the groaning networks up and running during the pandemic. There are a million variables. But for those using chaos-engineering techniques—even for as far-flung and traditional a business as DBS Bank, a consumer and investment institution in Singapore with US \$437 billion in assets—they helped. DBS is three years into a network resiliency program, site-reliability engineer Harpreet Singh says, and even as the program got off the ground in early 2018 the team was experimenting with chaos-engineering tools.

And chaos seems to be catching. Jones's [Jeli startup delivers a strategic view](#) on what she calls the catalyst events (events that might be simulated or sparked by chaos engineering), which show the difference between how an organization thinks it works and how it actually works. Gremlin, a four-year-old San Jose venture, offers chaos-engineering tools as service products. In January, the company also issued its first "State of Chaos Engineering" report for 2021. In a blog post announcing the publication, Gremlin vice president of marketing Aileen Horgan described chaos-engineering conferences these days as topping 3,500-plus registrants. Gremlin's user base alone, she noted, has conducted nearly 500,000 chaos-engineering system attacks to date.

Gonzalez says AWS has been using chaos-engineering practices for a long time. This year—as the networked world, hopefully, recovers from stress that tested it like never before—AWS is launching a [fault-injection service](#) for its cloud customers to use to run their own experiments.

Who knows how they'll be needed.