### principles of chaos engineering

进行混沌实验的一般方法：

* 假设稳定状态（定义什么是稳定状态）
* 检测真实事件（监控系统真实数据）
* 运行混沌测试
* 自动化运行混沌测试

##### 1. 假设稳定状态

混沌工程的最终目的时通过混沌测试，以提高系统的稳定性、可用性。所以首先我们需要定义系统的什么状态是合理运行的状态。其中我们最关心的是复杂系统的**可用性**。因此我们需要监控系统的多个指标，以期能准确地刻画系统的运行状态。

在微服务系统中，整个项目由大量的单独服务组成。每个服务都有可能失效故障。而在这些服务当值，有一些服务的失效不会影响系统整体的可用性

如Netflix网站中有用户自定义系统，如根据用户浏览历史的推荐服务；用户关注的排名服务；历史浏览的进度条保存服务，数据缓存服务等。这些服务属于对整个系统的稳定性，可用性不太重要的服务，即这些服务的失效不会导致系统的严重问题。

因此我们在设定系统稳定状态时，要找到影响系统可用性，稳定性的关键指标度量。

如在Netflix网站中，我们更关心的是用户能否准确找到需要的内容，能否流畅的观看视频等。

因此我们规定了一个度量指标：SPS（Start Per Second）即对于同一个视频，每秒有多少用户点击播放了该视频。

我们使用SPS作为监控系统健康状况的首要指标。混沌工程中的”混沌“有不确定的意思。混沌工程的基本思想是假定系统的监测数据是规律的，可大致预测的。

网飞定义的另一个系统监控指标是signup per second。选择signups per second，start per second这两个度量指标的意义在于，根据这两个度量指标可以很容易的感知到系统的注册服务，视频播放服务有无故障。

如果用户不能注册，登录，播放视频，则系统明显出现故障了

其他的服务可能会使用其他不同的度量指标刻画服务的可用性。如在线交易服务的度量指标可能是completed purchases per second；广告服务的度量指标可能是ads viewed by users per second。

相较于SPS指标以及CPU使用率，数据库查表耗时等度量指标。混沌工程更偏向于使用前者度量指标。因为再次，我们更关注于”应用“的稳定性。

后者CPU负载，数据库查表耗时等指标更偏向于刻画集群，分布式系统底层组件的可用性。处于更底层的层面。

##### 2. 监控真实场景

<https://prometheus.io/>

进行混沌测试时，尽量使用在真实场景中出现的数据。容易想到，使用系统运行的历史数据来进行混沌测试。

同时，任何打破系统稳定状态的数据都可用来作为混沌测试的数据。

下面是网飞使用的一些数据：

* 终止虚拟机实例
* 注入请求延迟
* 注入服务间请求失败
* 内部服务失效
* 集群部分失效

##### 3. 运行混沌测试