In chaos engineering, a "CPU pod hog" scenario refers to deliberately creating a situation where one or more pods (containers) within a Kubernetes cluster consume an excessive amount of CPU resources. This action is intended to simulate real-world scenarios where certain processes or applications suddenly demand more CPU resources than expected, potentially leading to performance degradation or service disruptions.

Kubectl top po

|  |  |  |
| --- | --- | --- |
| **Variables** | **Description** | **Notes** |
| CPU\_CORES | Number of the cpu cores subjected to CPU stress | Default to 1 |
| TOTAL\_CHAOS\_DURATION | The time duration for chaos insertion (seconds) | Default to 60s |
| LIB | The chaos lib used to inject the chaos. Available libs are litmus and pumba | Default to litmus |
| LIB\_IMAGE | Image used to run the helper pod. | Defaults to litmuschaos/go-runner:1.13.8 |
| STRESS\_IMAGE | Container run on the node at runtime by the pumba lib to inject stressors. Only used in LIB pumba | Default to alexeiled/stress-ng:latest-ubuntu |
| TARGET\_PODS | Comma separated list of application pod name subjected to pod cpu hog chaos | If not provided, it will select target pods randomly based on provided appLabels |
| TARGET\_CONTAINER | Name of the target container under chaos | If not provided, it will select the first container of the target pod |
| PODS\_AFFECTED\_PERC | The Percentage of total pods to target | Defaults to 0 (corresponds to 1 replica), provide numeric value only |
| CONTAINER\_RUNTIME | container runtime interface for the cluster | Defaults to containerd, supported values: docker, containerd and crio for litmus and only docker for pumba LIB |
| SOCKET\_PATH | Path of the containerd/crio/docker socket file | Defaults to /run/containerd/containerd.sock |
| RAMP\_TIME | Period to wait before injection of chaos in sec |  |
| SEQUENCE | It defines sequence of chaos execution for multiple target pods | Default value: parallel. Supported: serial, parallel |