



2018.3.21

Outsider @ Korea Chaos Engineering Meetup #3



While overlooking Netflix's migration to the cloud in 2011, Greg Orzell had the idea to change this paradigm by setting up a tool that would cause breakdowns in their production environment, the environment used by Netflix customers.



**The code behind Chaos Monkey was
released by Netflix in 2012
under an Apache 2.0 license.**



2016년 2.0 릴리스



This version of Chaos Monkey is fully integrated with Spinnaker, the continuous delivery platform that we use at Netflix. You must be managing your apps with Spinnaker to use Chaos Monkey to terminate instances.



Chaos Monkey randomly terminates virtual machine instances and containers that run inside of your production environment.

Exposing engineers to failures more frequently incentivizes them to build resilient services.



**an open source, multi-cloud continuous
delivery platform for releasing software
changes with high velocity and confidence.**

Spinnaker Chart

[Spinnaker](#) is an open source, multi-cloud continuous delivery platform.

Chart Details

This chart will provision a fully functional and fully featured Spinnaker installation that can deploy and manage applications in the cluster that it is deployed to.

Redis and Minio are used as the stores for Spinnaker state.

For more information on Spinnaker and its capabilities, see it's [documentation](#).

Installing the Chart

To install the chart with the release name `my-release` :

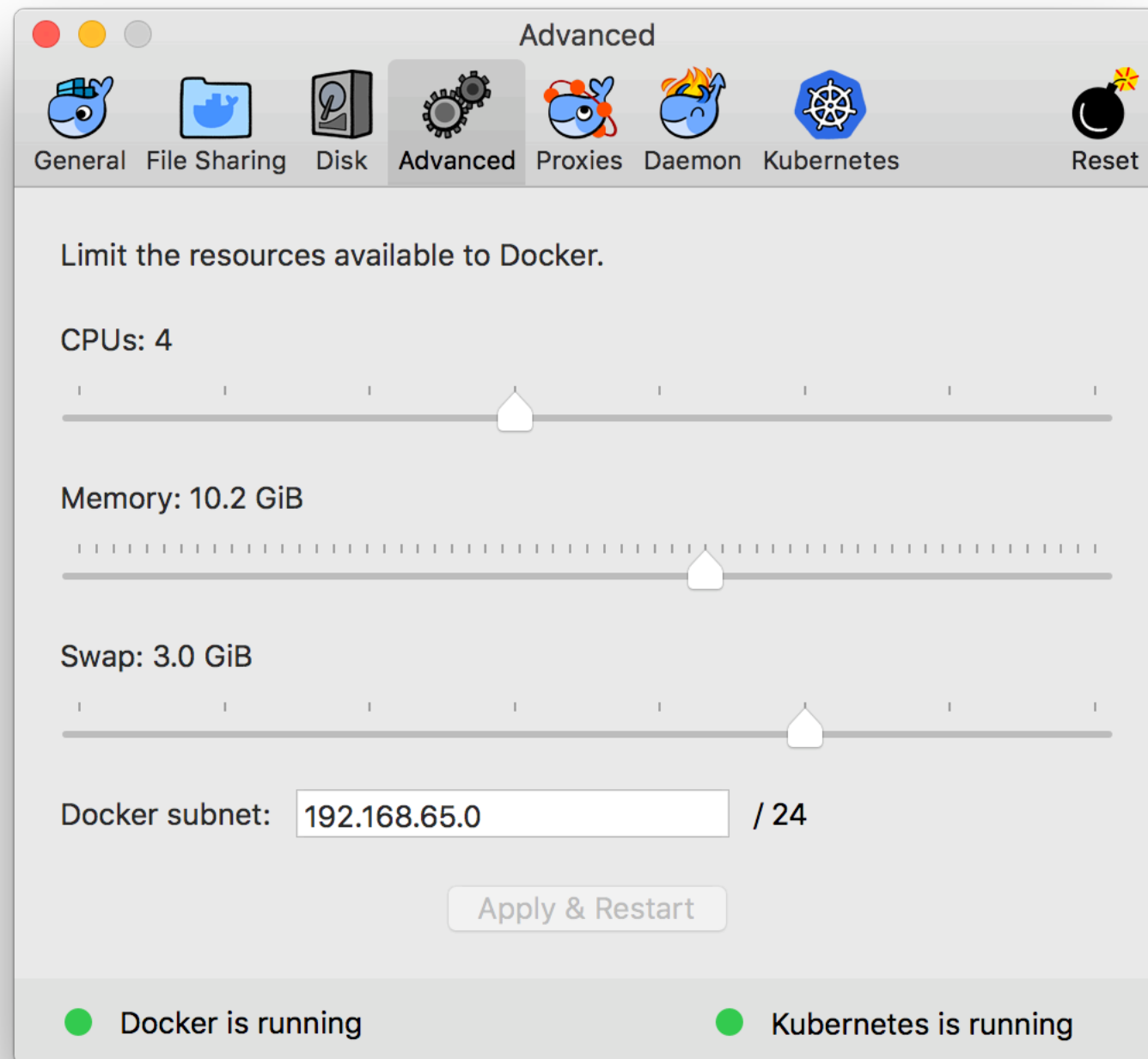
```
$ helm install --name my-release stable/spinnaker
```

Note that this chart pulls in many different Docker images so can take a while to fully install.

**Kubernetes Helm : Kubernetes charts를
관리하는 도구, charts는 미리 구성된 Kubernetes
리소스의 패키지다.**

```
$ brew install kubernetes-helm
```

```
$ helm init
```



```
$ curl -Lo values.yaml \  
  https://raw.githubusercontent.com/kubernetes/charts/master/  
stable/spinnaker/values.yaml
```

```
$ helm install --name chaos ./spinnaker -f values.yaml --timeout 600
```

NAME: chaos

LAST DEPLOYED: Wed Mar 21 03:42:29 2018

NAMESPACE: default

STATUS: DEPLOYED

RESOURCES:

==> v1/Secret

NAME	TYPE	DATA	AGE
chaos-jenkins	Opaque	2	4m
chaos-minio-user	Opaque	2	4m
chaos-redis	Opaque	1	4m
chaos-spinnaker-registry	Opaque	1	4m

==> v1/ConfigMap

NAME	DATA	AGE
chaos-jenkins	3	4m
chaos-jenkins-tests	1	4m
chaos-minio-config-cm	1	4m
chaos-jenkins-jobs	3	4m
chaos-spinnaker-s3-config	1	4m
chaos-spinnaker-spinnaker-config	18	4m
chaos-spinnaker-tests	1	4m

==> v1/PersistentVolumeClaim

NAME	STATUS	VOLUME	CAPACITY	ACCESS MODES	STORAGECLASS	AGE
chaos-jenkins	Bound	pvc-71c57b2a-2c6e-11e8-ae5d-025000000001	8Gi	RWO	hostpath	4m
chaos-minio	Bound	pvc-71c5d751-2c6e-11e8-ae5d-025000000001	10Gi	RWO	hostpath	4m
chaos-redis	Bound	pvc-71c635ad-2c6e-11e8-ae5d-025000000001	8Gi	RWO	hostpath	4m

==> v1/Service

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
chaos-jenkins-agent	ClusterIP	10.101.82.20	<none>	50000/TCP	4m
chaos-jenkins	ClusterIP	10.110.229.154	<none>	8080/TCP	4m
chaos-minio-svc	ClusterIP	10.103.163.71	<none>	9000/TCP	4m
chaos-redis	ClusterIP	10.98.101.239	<none>	6379/TCP	4m
chaos-spinnaker-clouddriver	ClusterIP	10.96.11.251	<none>	7002/TCP	4m
chaos-spinnaker-deck	ClusterIP	10.106.78.204	<none>	9000/TCP	4m
chaos-spinnaker-echo	ClusterIP	10.104.108.246	<none>	8089/TCP	4m



Search

projects, applications, clusters, load balancers, server groups, security groups

Actions ▾

Recently viewed



APPLICATIONS (1)

nginx



- 애플리케이션 생성
- 로드밸런서 생성
- 서버그룹 생성

```
$ go get github.com/netflix/chaosmonkey/cmd/chaosmonkey
```

```
$ chaosmonkey --version  
2.0.2
```

```
$ docker run -d -p 3306:3306 --name mysql -e  
MYSQL_ROOT_PASSWORD=password mysql
```

```
$ docker run -it --link mysql:mysql --rm mysql \  
sh -c 'exec mysql -h"$MYSQL_PORT_3306_TCP_ADDR" \  
-P"$MYSQL_PORT_3306_TCP_PORT" -uroot -ppassword'
```

```
Welcome to the MySQL monitor.  Commands end with ; or \g.  
Your MySQL connection id is 2  
Server version: 5.7.21 MySQL Community Server (GPL)
```

```
Copyright (c) 2000, 2018, Oracle and/or its affiliates. All  
rights reserved.
```

```
Oracle is a registered trademark of Oracle Corporation and/or  
its  
affiliates. Other names may be trademarks of their respective  
owners.
```

```
Type 'help;' or '\h' for help. Type '\c' to clear the current  
input statement.
```

```
mysql> CREATE DATABASE chaosmonkey;  
Query OK, 1 row affected (0.00 sec)
```

chaosmonkey.toml

```
[chaosmonkey]
enabled = true
schedule_enabled = true
leashed = false
accounts = ["local"]

[database]
host = "127.0.0.1"
name = "chaosmonkey"
user = "root"
encrypted_password = "password"

[spinnaker]
endpoint = "http://localhost:9000/gate"
```

chaosmonkey.toml

```
$ chaosmonkey migrate  
[30897] 2018/03/21 04:06:03 Successfully applied database  
migrations. Number of migrations applied: 1  
[30897] 2018/03/21 04:06:03 database migration applied  
successfully
```

```
$ chaosmonkey config nginx
```

```
(*chaosmonkey.AppConfig)(0xc4201ee1c0)({  
  Enabled: (bool) true,  
  RegionsAreIndependent: (bool) true,  
  MeanTimeBetweenKillsInWorkDays: (int) 2,  
  MinTimeBetweenKillsInWorkDays: (int) 1,  
  Grouping: (chaosmonkey.Group) cluster,  
  Exceptions: ([]chaosmonkey.Exception) {  
  },  
  Whitelist: (*[]chaosmonkey.Exception)(<nil>)  
})
```



```
$ chaosmonkey terminate nginx local --cluster=nginx-prod
```

NETFLIX

<https://github.com/Netflix/SimianArmy/blob/master/assets/SimianArmy.png>



SimianArmy <https://github.com/Netflix/SimianArmy>

Chaos Monkey : Netflix가 만든 첫번째 도구로 프로덕션 환경에서 랜덤으로 고
른 인스턴스를 죽인다.

Chaos Gorilla : SimianArmy 계층 맨 위에 있으면 Amazon AZ 전체를
죽인다.

Latency Monkey: 네트워크의 성능을 낮추거나 동작하지 않도록 시뮬레이션 해서
통신을 지연시킨다.

Doctor Monkey: 근본 원인 분석과 수정 여부에 대한 언헬시한 인스턴스를 탐지하
기 위해 CPU 부하등 성능 매트릭을 모니터링하고 헬스체크를 한다.

Janitor Monkey: 낭비를 줄이기 위해 사용하지 않는 리소스를 찾는다.

Conformity Monkey: 규칙을 적용해서 인스턴스가 규칙을 따르는지 판단하는 도
구다.

Security Monkey: Conformity Monkey에서 분기된 도구로 알려진 취약점이
있거나 부적절한 설정이 있는 인스턴스를 찾아서 비활성화한다.

10-18 Monkey: 다른 지역의 리전간에 고객에게 제공하는 소프트웨어에서 I10n -
i18n 문제를 찾는 도구.