**Node affinity란?**

Node affinity는 Pod가 특정 node로 배포되도록 하는 기능이다. 예전에 label에 설명했던 node selector도 node의 label과 pod의 selector label이 매칭되는 node에만 배포하도록 하기 때문에, 사실상 Node affinity와 같은 기능을한다.

Node affinity는 Hard affinity와 Soft Affinity가 있다. Node Affinity는 Pod가 조건에 딱 맞는 node에만 배포 되도록하는 기능이고 Soft Affinity는 Pod게 조건에 맞는 node에 되도록(반드시가 아니라)이면 배포되도록 하는 기능이다. 앞에서 언급한 node selector는 Hard Affinity에 해당한다.

|  |
| --- |
| apiVersion: v1  kind: Pod  metadata:    name: with-node-affinity  spec:    affinity:  **nodeAffinity:**        requiredDuringSchedulingIgnoredDuringExecution:          nodeSelectorTerms:          - matchExpressions:            - key: kubernetes.io/e2e-az-name              operator: In              values:              - e2e-az1              - e2e-az2        preferredDuringSchedulingIgnoredDuringExecution:        - weight: 1          preference:            matchExpressions:            - key: another-node-label-key              operator: In              values:              - another-node-label-value    containers:    - name: with-node-affinity      image: k8s.gcr.io/pause:2.0 |

requiredDuringSchedulingIgnoredDuringExecution는 Hard Affinity 정의이다. nodeSelectorTerms 부분에, matchExpressions를 사용해, label set-based selector 문법을 이용하면 된다. 위 예제는 node에 label key "kubernetes.io/e2e-az-name"의 값이 eze-az1이나 eze-az2인 node를 선택하도록하는 설정이다.

node affinity는 여러 affinity를 동시에 적용할 수 있는데, 위의 예제는 두 개의 Affinity를 정의했다. 두 번째 Affinity는 soft affinity로 preferredDuringSchedulingIgnoredDuringExecution으로 정의한다. Soft affinity는 앞서 언급한 것과 같이 조건에 맞는 node로 되도록이면 배포될 수 있도록 그 node로 배포 선호도를 주는 기능이다. 이때 weight 필드를 이용해 선호도를 조정할 수 있는데, weight는 1~100이고, node의 soft affinity의 weight 값들을 합쳐서 그 값이 높은 node를 우선으로 고려하도록 우선 순위를 주는데 사용할 수 있다.

특정 Nopde로 배포되게하는 Affinity 설정도 있으나, 반대로 특정 node로 배포되는 것을 피하도록 하는 AntiAffinity라는 설정도 있다. nodeAffinity대신 nodeAntiAffinity라는 Notation을사용하면 되고 Affinity와는 다르게 반대로, 조건에 맞는 Node를 피해서 배포하도록 된다.