

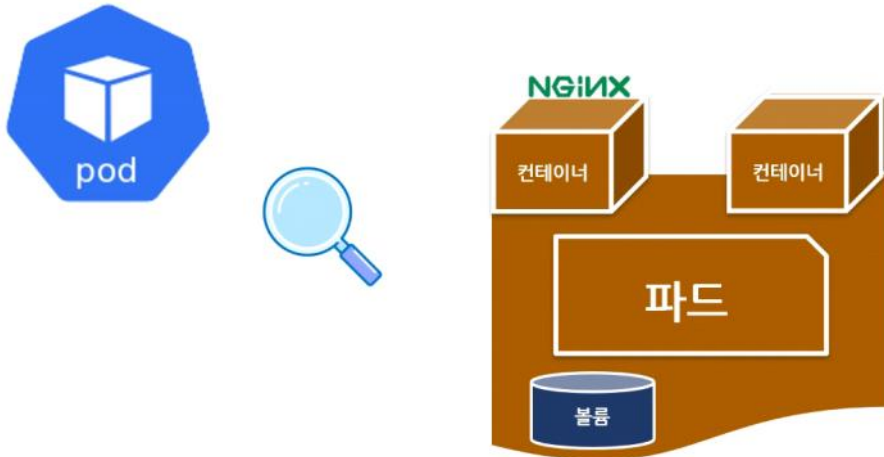
## 2\_ 배포를 통한 쿠버네티스 체험

2021년 11월 17일 수요일 오전 1:35

### 2.1 배포를 통해 확인하는 파드(pod)

파드란?

## 파드(Pod)란?



-> 컨테이너들의 집합

파드배포실습

```
kubectl run nginx --image=nginx
```

```
[root@m-k8s ~]# kubectl run nginx --image=nginx
pod/nginx created
[root@m-k8s ~]#
```

```
[root@m-k8s ~]# clear

[root@m-k8s ~]# kubectl run nginx --image=nginx
pod/nginx created
[root@m-k8s ~]# kubectl get pod
NAME                                READY   STATUS    RESTARTS   AGE
deploy-nginx-8458f6dbbb-2rbjx      1/1     Running   0           3d21h
deploy-nginx-8458f6dbbb-79qtb      1/1     Running   0           3d20h
deploy-nginx-8458f6dbbb-7nmr2      1/1     Running   0           3d20h
nginx                               1/1     Running   0           22s
[root@m-k8s ~]# kubectl get pod -o wide
NAME                                READY   STATUS    RESTARTS   AGE   IP             NODE       NOMINATED
NODE   READINESS GATES
deploy-nginx-8458f6dbbb-2rbjx      1/1     Running   0           3d21h  172.16.132.3   w3-k8s    <none>
deploy-nginx-8458f6dbbb-79qtb      1/1     Running   0           3d20h  172.16.132.4   w3-k8s    <none>
deploy-nginx-8458f6dbbb-7nmr2      1/1     Running   0           3d20h  172.16.103.129 w2-k8s    <none>
nginx                               1/1     Running   0           31s    172.16.221.130 w1-k8s    <none>
[root@m-k8s ~]# curl 172.16.221.130
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
html { color-scheme: light dark; }
body { width: 35em; margin: 0 auto;
font-family: Tahoma, Verdana, Arial, sans-serif; }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

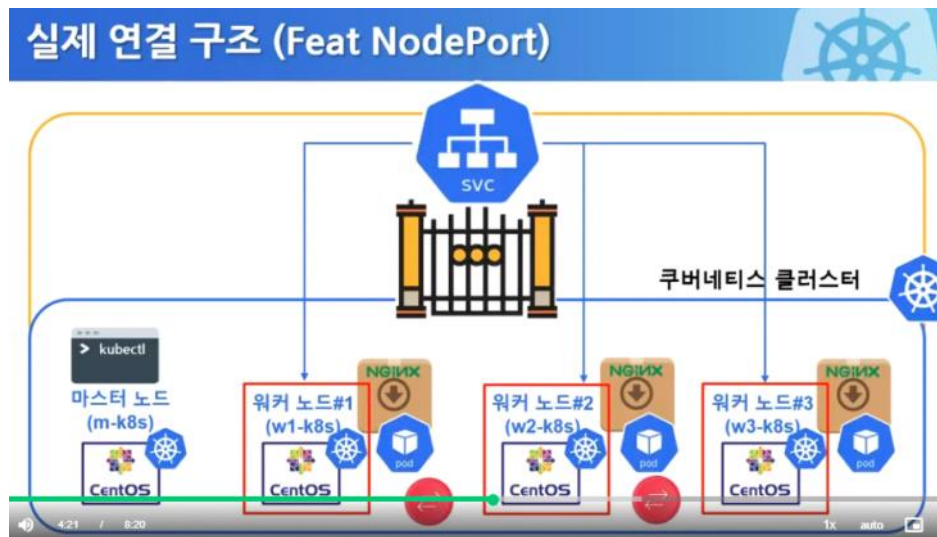
<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
[root@m-k8s ~]#
```

## 2.2 파드를 외부에서도 접속하게 하는 서비스( Service )

pod의 ip는 외부에서 접속이 안됨

쿠버네티스 클러스터가 외부에서 접속하기 위해서는 서비스를 통해야 함



생성된 pod를 Service로 연결

-> 명령어 : kubectl expose pod nginx --type=NodePort --port=80

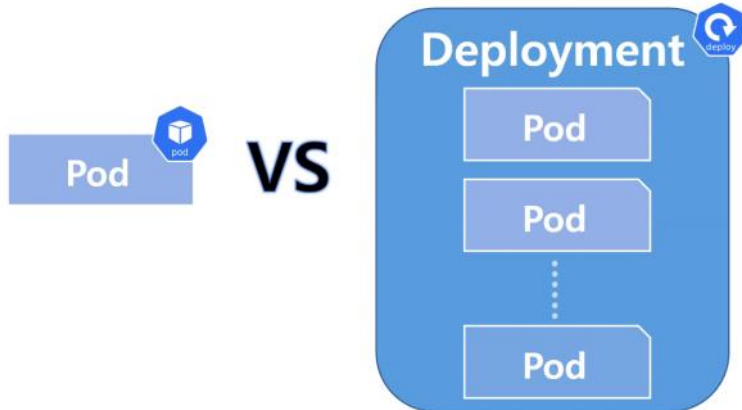
```
[root@m-k8s ~]# kubectl expose pod nginx --type=NodePort --port=80
service/nginx exposed
[root@m-k8s ~]#
```

## 2.3.파드와 디플로이먼트(Deployment) 차이

파드 한 개만 배포 -> 이슈있음

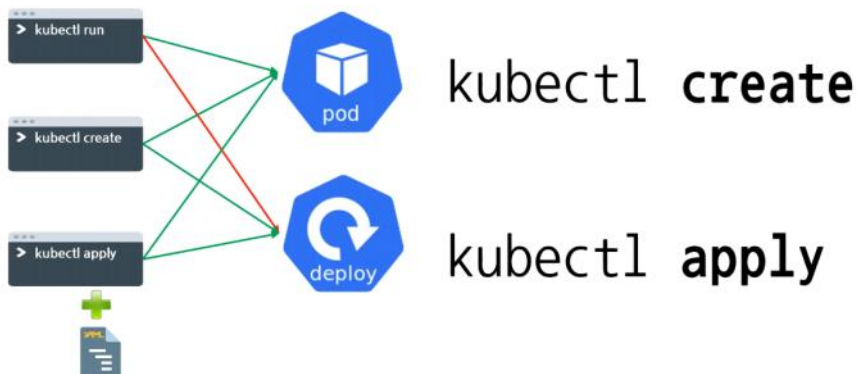
파드가 여러 개 배포가 되는 구조 : Deployment

## 파드와 디플로이먼트 비교



디플로이먼트 배포 방법

쿠버네티스 1.16 ~ 1.17버전까지는 Deployment 배포가 가능  
1.18버전부터는 kubectl run으로는 불가능  
kubectl create or kubectl apply로 배포가능



명령어 : kubectl create deployment deploy-nginx --image=nginx

```
[root@m-k8s ~]# kubectl get deployments
NAME          READY   UP-TO-DATE   AVAILABLE   AGE
deploy-nginx   3/3     3             3           3d21h
[root@m-k8s ~]#
```

디플로이먼트로 다수의 파드 배포

-> Replicaset 사용

파드의 배포수 늘리기

-> kubectl scale deployment deploy-nginx --replicas=3

```
[root@m-k8s ~]# kubectl get pod
NAME                                READY   STATUS    RESTARTS   AGE
deploy-nginx-8458f6dbbb-2rbjx       1/1     Running   0          3d21h
deploy-nginx-8458f6dbbb-79qtb       1/1     Running   0          3d21h
deploy-nginx-8458f6dbbb-7nmr2       1/1     Running   0          3d21h
nginx                                1/1     Running   0          22m
```

2.4.외부로 노출하는 더 좋은 방법인 로드밸런서(LoadBalancer)

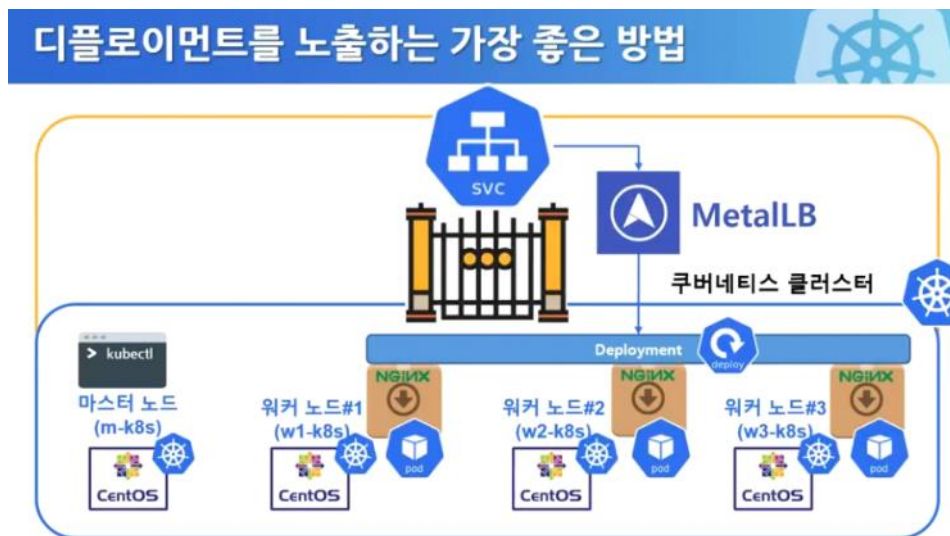
디플로이먼트를 외부로 노출( NodePort )

-> 명령어 `kubectl expose deployment deploy-nginx --type=NodePort --port=80`

```
[root@m-k8s ~]# kubectl get service
NAME          TYPE          CLUSTER-IP      EXTERNAL-IP      PORT(S)
AGE
deploy-nginx   NodePort       10.101.25.166    <none>           80:32247/TCP
14s
kubernetes     ClusterIP      10.96.0.1        <none>           443/TCP
4d23h
nginx          NodePort       10.106.43.93     <none>           80:30335/TCP
17m
[root@m-k8s ~]# kubectl expose deployment deploy-nginx --type=NodePort
--port=80
```

디플로이먼트를 외부로 노출( LoadBalancer )

MetalLB를 선언하여 접속



노드포트보다 로드밸런서가 좋은 점

- > 대표 IP를 노출함으로써 보안상 강점
- > 경로의 최적화

MetalLB 설치 : `kubectl apply -f ~/Lecture_k8s_starter.kit/ch2/2.4/metallb.yaml`

```
[root@m-k8s ~]# kubectl apply -f ~/Lecture_k8s_starter.kit/ch2/2.4/metallb.yaml
namespace/metallb-system created
podsecuritypolicy.policy/speaker created
serviceaccount/controller created
serviceaccount/speaker created
clusterrole.rbac.authorization.k8s.io/metallb-system:controller created
clusterrole.rbac.authorization.k8s.io/metallb-system:speaker created
role.rbac.authorization.k8s.io/config-watcher created
clusterrolebinding.rbac.authorization.k8s.io/metallb-system:controller created
clusterrolebinding.rbac.authorization.k8s.io/metallb-system:speaker created
rolebinding.rbac.authorization.k8s.io/config-watcher created
daemonset.apps/speaker created
deployment.apps/controller created
configmap/config created
[root@m-k8s ~]#
```

마스터노드에 이미 metallb.yaml 파일이 다운로드 되어 있음

디플로이먼트 생성 : `kubectl create deployment chk-hn --image=sysnet4admin/chk-hn`

생성된 디플로이먼트내의 pod 수 증가 : `kubectl scale deployment chk-hn --replicas=3`

```
[root@m-k8s ~]# kubectl create deployment chk-hn --image=sysnet4admin/chk-hn
deployment.apps/chk-hn created
[root@m-k8s ~]# kubectl scale deployment chk-hn --replicas=3
deployment.apps/chk-hn scaled
```

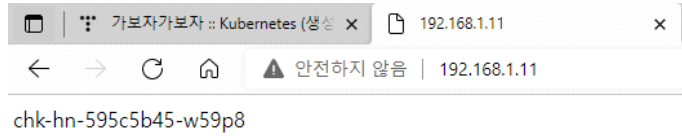
로드밸런서 타입으로 디플로이먼트 외부에 노출

-> 명령어 : kubectl expose deployment chk-hn --type=LoadBalancer --port=80

```
[root@m-k8s ~]# kubectl expose deployment chk-hn --type=LoadBalancer --port=80
service/chk-hn exposed
[root@m-k8s ~]# kubectl get services
```

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
chk-hn	LoadBalancer	10.97.177.111	192.168.1.11	80:31145/TCP	8s
deploy-nginx	NodePort	10.101.25.166	<none>	80:32247/TCP	12m
kubernetes	ClusterIP	10.96.0.1	<none>	443/TCP	4d23h
nginx	NodePort	10.106.43.93	<none>	80:30335/TCP	29m

```
[root@m-k8s ~]#
```



```
[root@m-k8s ~]# kubectl get pods -o wide
```

NAME	READY	STATUS	RESTARTS	AGE	IP	NODE	NOMINATED	NODE	READINESS	GATES
chk-hn-595c5b45-bgwjz	1/1	Running	0	4m34s	172.16.132.5	w3-k8s	<none>		<none>	
chk-hn-595c5b45-crcdz	1/1	Running	0	4m34s	172.16.103.131	w2-k8s	<none>		<none>	
chk-hn-595c5b45-w59p8	1/1	Running	0	5m13s	172.16.221.131	w1-k8s	<none>		<none>	
deploy-nginx-8458f6dbbb-2rbjx	1/1	Running	0	3d21h	172.16.132.3	w3-k8s	<none>		<none>	
deploy-nginx-8458f6dbbb-79gtb	1/1	Running	0	3d21h	172.16.132.4	w3-k8s	<none>		<none>	
deploy-nginx-8458f6dbbb-7nmr2	1/1	Running	0	3d21h	172.16.103.129	w2-k8s	<none>		<none>	
nginx	1/1	Running	0	39m	172.16.221.130	w1-k8s	<none>		<none>	

```
[root@m-k8s ~]#
```

## 2.5. 배포한 것들 삭제하기

-> 명령어 : kubectl delete service < 서비스명 >

```
[root@m-k8s ~]# kubectl delete service chk-hn
service "chk-hn" deleted
[root@m-k8s ~]# kubectl get service
```

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
deploy-nginx	NodePort	10.101.25.166	<none>	80:32247/TCP	16m
kubernetes	ClusterIP	10.96.0.1	<none>	443/TCP	4d23h
nginx	NodePort	10.106.43.93	<none>	80:30335/TCP	33m

```
[root@m-k8s ~]# kubectl delete service deploy-nginx
service "deploy-nginx" deleted
[root@m-k8s ~]# kubectl delete service nginx
service "nginx" deleted
[root@m-k8s ~]#
```

-> 명령어 : kubectl delete deployment < 디플로이먼트명 >

```
[root@m-k8s ~]# kubectl get deployment
```

NAME	READY	UP-TO-DATE	AVAILABLE	AGE
chk-hn	3/3	3	3	10m
deploy-nginx	3/3	3	3	3d21h

```
[root@m-k8s ~]# kubectl delete deployment chk-hn
deployment.apps "chk-hn" deleted
[root@m-k8s ~]# kubectl delete deployment deploy-nginx
deployment.apps "deploy-nginx" deleted
[root@m-k8s ~]# kubectl get deployment
No resources found in default namespace.
[root@m-k8s ~]#
```