SERVICES

Problem Statement:

- ➔ *If load increases, we can create one more Pod. Who will create it ?*
- ➔ *There are two ways: manual or replication controller (it will set the desire state for you automatically)*
- ➔ *So, we cannot give all IP address of servers to the client -> not user friendly.*
- ➔ *We are going to create an intermediate program between client and pods. Say it has the IP (100), now client comes to this ip 100 and the request is recreated and connect to respective port of backend servers.*
- ➔ *This intermediate program is frontend of the backend servers and is also known as LOAD BALANCER.*

Now challenge is, how LB will register as the new pod launches.

- ➔ *Since IP changes of system on each restart, so we have to tag the OS or label it.*
- ➔ *LB will look for this Label and as soon as it finds it, that particular OS will get register under LB and will be reflected in ENDPOINTS of LB.*

Terminologies:

- ➔ Cluster IP: load balancing with system
- ➔ NodePort: load balancing within public world and the program used is known as KUBEPROXY.
- ➔ External LB: when we use LB of any other cloud services, or manually created LB in K8

1. Create a LB code

2. Launch pod(lbpod1) and register under Endpoint of LB

3. Launch another pod(lbpod2) and register under Endpoint of LB

4.Verifying CLuster IP Load balancing.

5. Exposing to Public World: NodePort Load balancing

6. Verifying NodePort Load balancing via CLI and GUI both.

1st:

```
apiVersion: v1
kind: Service

metadata:
  name: mylb1

spec:
 selector:
   app: hacker

 ports:
  - targetPort: 80
    port: 8080
```

kubectl create -f svc.yml - *create the mylb1*

```
C:\Users\Romio_juliete\Desktop\CKA_ws_akshayanil>kubectl create -f svc.yml
service/mylb1 created
```

kubectl get svc - *status of services*

```
C:\Users\Romio_juliete\Desktop\CKA_ws_akshayanil>kubectl get svc
NAME         TYPE        CLUSTER-IP       EXTERNAL-IP   PORT(S)        AGE
kubernetes   ClusterIP   10.96.0.1        <none>        443/TCP        47h
mylb1        ClusterIP   10.104.92.171    <none>        8080/TCP       15m
myrc1        ClusterIP   10.101.46.85     <none>        80/TCP         3h14m
myrc2        NodePort    10.103.166.104   <none>        80:31504/TCP   175m
```

kubectl describe svc mylb1

```
C:\Users\Romio_juliete\Desktop\CKA_ws_akshayanil>kubectl describe svc mylb1
Name:                     mylb1
Namespace:                default
Labels:                   <none>
Annotations:              <none>
Selector:                 app=hacker
Type:                     ClusterIP
IP Families:              <none>
IP:                       10.104.92.171
IPs:                      10.104.92.171
Port:                     <unset>  8080/TCP
TargetPort:               80/TCP
Endpoints:                <none>
Session Affinity:         None
Events:                   <none>
```

Now, it has no backend servers as endpoints is none…

2nd:

Launching one pod with label web: hacker

Code:

lbpod - Notepad

File Edit Format View Help

```
apiVersion: v1
kind: Pod
metadata:
 name: "LBpod1"
 labels:
   app: hacker

spec:
 containers:
 - name: "myc1"
   image: "vimal13/apache-webserver-php"
```

kubectl create -f lbpod.yml

```
C:\Users\Romio_juliete\Desktop\CKA_ws_akshayanil>kubectl create -f lbpod.yml
pod/lbpod1 created
```

```
C:\Users\Romio_juliete\Desktop\CKA_ws_akshayanil>kubectl describe pods lbpod1
Name:         lbpod1
Namespace:    default
Priority:     0
Node:         minikube/192.168.99.101
Start Time:   Fri, 15 Jan 2021 21:23:18 +0530
Labels:       app=hacker
Annotations:  <none>
Status:       Running
IP:           172.17.0.10
```

Check the endpoints in mylb1

kubectl describe svc mylb1

```
C:\Users\Romio_juliete\Desktop\CKA_ws_akshayanil>kubectl describe svc mylb1
Name:                     mylb1
Namespace:                default
Labels:                   <none>
Annotations:              <none>
Selector:                 app=hacker
Type:                     ClusterIP
IP Families:              <none>
IP:                       10.104.92.171
IPs:                      10.104.92.171
Port:                     <unset>  8080/TCP
TargetPort:               80/TCP
Endpoints:                172.17.0.10:80
Session Affinity:         None
Events:                   <none>
```

FANATSTIC…….. The IP of above pod launched is added in the endpoints in LB.

3rd:

Now let's add a new pod with same labels to add one more endpoint to LB.

Code:

```
*lbpod - Notepad
File  Edit  Format  View  Help
apiVersion: v1
kind: Pod
metadata:
 name: "lbpod2"
 labels:
   app: hacker

spec:
 containers:
 - name: "myc1"
   image: "vimal13/apache-webserver-php"
```

Ln 4, Col 15 | 100% | Windows (CRLF) | UTF-8

kubectl apply -f lbpod.yml

```
C:\Users\Romio_juliete\Desktop\CKA_ws_akshayanil>kubectl apply -f lbpod.yml
pod/lbpod2 created
```

```
C:\Users\Romio_juliete\Desktop\CKA_ws_akshayanil>kubectl describe pods lbpod2
Name:         lbpod2
Namespace:    default
Priority:     0
Node:         minikube/192.168.99.101
Start Time:   Fri, 15 Jan 2021 21:32:06 +0530
Labels:       app=hacker
Annotations:  <none>
Status:       Running
IP:           172.17.0.11
```

Check the endpoints in mylb1- it should have now two endpoints.

kubectl describe svc mylb1

```
C:\Users\Romio_juliete\Desktop\CKA_ws_akshayanil>kubectl describe svc mylb1
Name:                     mylb1
Namespace:                default
Labels:                   <none>
Annotations:              <none>
Selector:                 app=hacker
Type:                     ClusterIP
IP Families:              <none>
IP:                       10.104.92.171
IPs:                      10.104.92.171
Port:                     <unset>  8080/TCP
TargetPort:               80/TCP
Endpoints:                172.17.0.10:80,172.17.0.11:80
Session Affinity:         None
Events:                   <none>
```

The above two endpoints are actually the IP of two pods (lbpod1, lbpod2) we launched above.

4th: docker container don't have the outside connectivity.

My k8 is running in VB(Minikube) , so the hosted server can be accessed within the minikube only for now.

```
C:\Users\Romio_juliete\Desktop\CKA_ws_akshayanil>kubectl get svc
NAME         TYPE        CLUSTER-IP       EXTERNAL-IP   PORT(S)    AGE
kubernetes   ClusterIP   10.96.0.1        <none>        443/TCP    47h
mylb1        ClusterIP   10.104.92.171    <none>        8080/TCP   15m
```

Let's verify:

Open Minikube and type the IP and port number of ; curl 10.104.92.171:8080

```
minikube [Running] - Oracle VM VirtualBox
File  Machine  View  Input  Devices  Help
</pre>
$ curl http://10.104.92.171:8080
<body bgcolor='aqua'>
<pre>

welcome to vimal web server for testingeth0: flags=4163<UP,BROADCAST,RUNNING,MUL
TICAST>  mtu 1500
        inet 172.17.0.11  netmask 255.255.0.0  broadcast 172.17.255.255
        ether 02:42:ac:11:00:0b  txqueuelen 0  (Ethernet)
        RX packets 5  bytes 372 (372.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 3  bytes 182 (182.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0


</pre>
$
Right Ctrl
```

Again and the Ip changes :

```
minikube [Running] - Oracle VM VirtualBox
File  Machine  View  Input  Devices  Help
</pre>
$ curl http://10.104.92.171:8080
<body bgcolor='aqua'>
<pre>

welcome to vimal web server for testingeth0: flags=4163<UP,BROADCAST,RUNNING,MUL
TICAST>  mtu 1500
        inet 172.17.0.10  netmask 255.255.0.0  broadcast 172.17.255.255
        ether 02:42:ac:11:00:0a  txqueuelen 0  (Ethernet)
        RX packets 19  bytes 1421 (1.3 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 13  bytes 2782 (2.7 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0


</pre>
$ curl http://10.104.92.171:8080
Right Ctrl
```

CONGO: LOAD BALANCER WORKING FINE WITHIN CLUSTER IP.

Lets check it on public world: ie. From any other system - it will not work…

```
C:\Users\Romio_juliete\Desktop\CKA_ws_akshayanil>curl http://10.104.92.171:8080
```

5th:

Let's expose this setup to public world

Ans: type:NodePort Load balanacer.

Edit the code:

type:NodePort under pod spec

nodePort: 30000 under container spec

svc5 - Notepad

File Edit Format View Help

```
apiVersion: v1
kind: Service

metadata:
  name: mylb1

spec:
 type: NodePort
 selector:
   app: hacker

 ports:
  - targetPort: 80
    port: 8080
    nodePort: 30000
```

kubectl apply -f svc5.yml

kubectl get svc

```
C:\Users\Romio_juliete\Desktop\CKA_ws_akshayanil>kubectl apply -f svc5.yml
service/mylb1 configured

C:\Users\Romio_juliete\Desktop\CKA_ws_akshayanil>kubectl get svc
NAME         TYPE        CLUSTER-IP       EXTERNAL-IP   PORT(S)          AGE
kubernetes   ClusterIP   10.96.0.1        <none>        443/TCP          2d
mylb1        NodePort    10.104.92.171    <none>        8080:30000/TCP   74m
myrc1        ClusterIP   10.101.46.85     <none>        80/TCP           4h13m
myrc2        NodePort    10.103.166.104   <none>        80:31504/TCP     3h54m
```

Since we are using NodePort(Minikube) service for Load balancing, therefore we have to use Node IP as a LB.

Let's check the minukube Ip. lfconfig | less

```
          TX packets:1408 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:415516 (405.7 KiB)  TX bytes:246322 (240.5 KiB)

eth1      Link encap:Ethernet  HWaddr 08:00:27:A4:3F:83
          inet addr:192.168.99.101  Bcast:192.168.99.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:2555 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1483 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:229047 (223.6 KiB)  TX bytes:3662343 (3.4 MiB)

standard input
```

So, final IP of LB is 192.168.99.101:30000

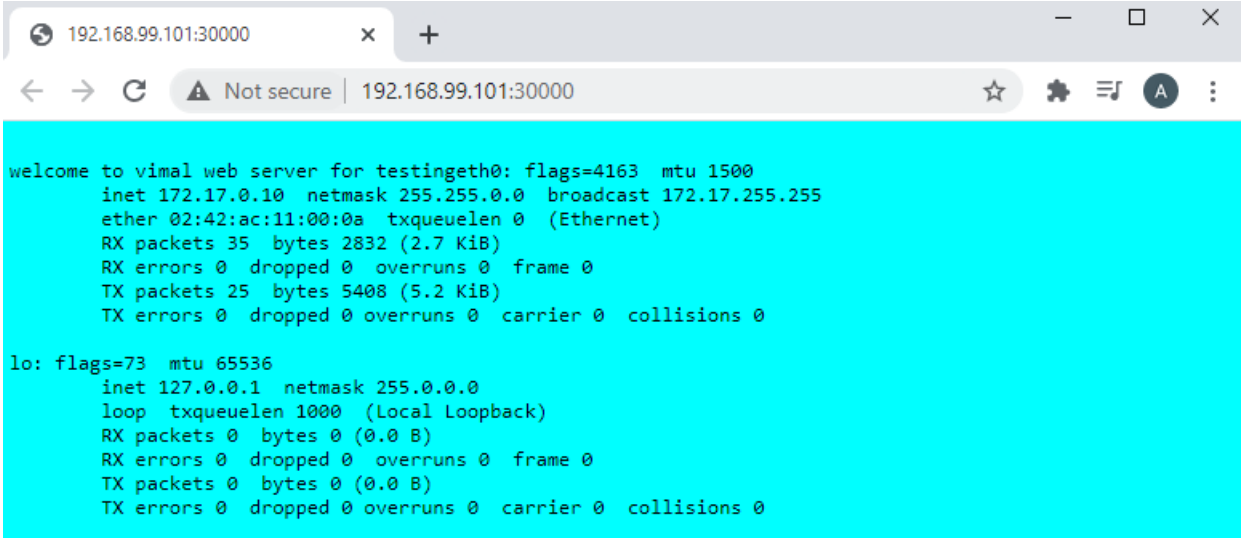
6th:

Verify with public world: say from my windows:

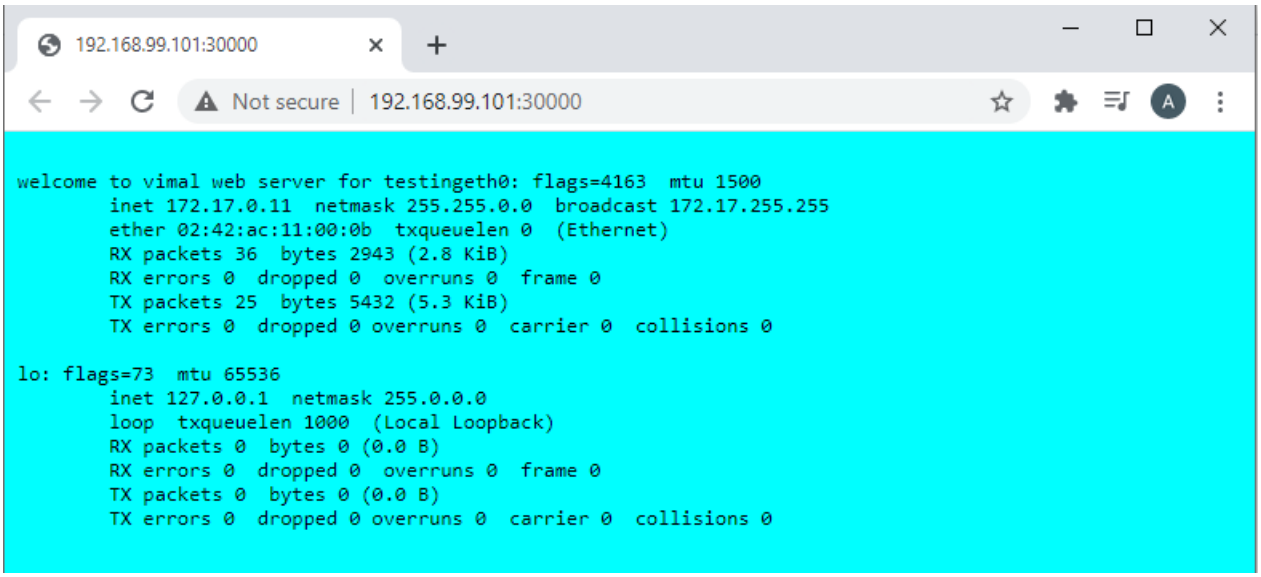
```
C:\Users\Romio_juliete\Desktop\CKA_ws_akshayanil>curl http://192.168.99.101:30000
<body bgcolor='aqua'>
<pre>

welcome to vimal web server for testingeth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 172.17.0.11  netmask 255.255.0.0  broadcast 172.17.255.255
        ether 02:42:ac:11:00:0b  txqueuelen 0  (Ethernet)
        RX packets 24  bytes 1842 (1.7 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 16  bytes 3981 (3.8 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0


</pre>

C:\Users\Romio_juliete\Desktop\CKA_ws_akshayanil>curl http://192.168.99.101:30000
<body bgcolor='aqua'>
<pre>

welcome to vimal web server for testingeth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 172.17.0.10  netmask 255.255.0.0  broadcast 172.17.255.255
        ether 02:42:ac:11:00:0a  txqueuelen 0  (Ethernet)
        RX packets 28  bytes 2027 (1.9 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 19  bytes 4107 (4.0 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0


</pre>

C:\Users\Romio_juliete\Desktop\CKA_ws_akshayanil>
```

FANTASTIC : working all fine and balancing the load also….

We can also see in gui mode…



Refresh…



Working fine……..

Finally setup the difference load balancer services in K8

- ➔ Cluster IP: load balancing with system
- ➔ NodePort: load balancing within public world.
- ➔ External LB: when we use LB of any other cloud services, or manually created LB in K8