

We have seen the NodePort service that helps us make an external facing application available on a port on the worker nodes. Let’s turn our focus to the front-end applications, which are the voting-app and the result-app. We know that these pods are hosted on the worker nodes in a cluster. Let’s say we have four node cluster and to make the applications accessible to external users, we create services of type NodePort.

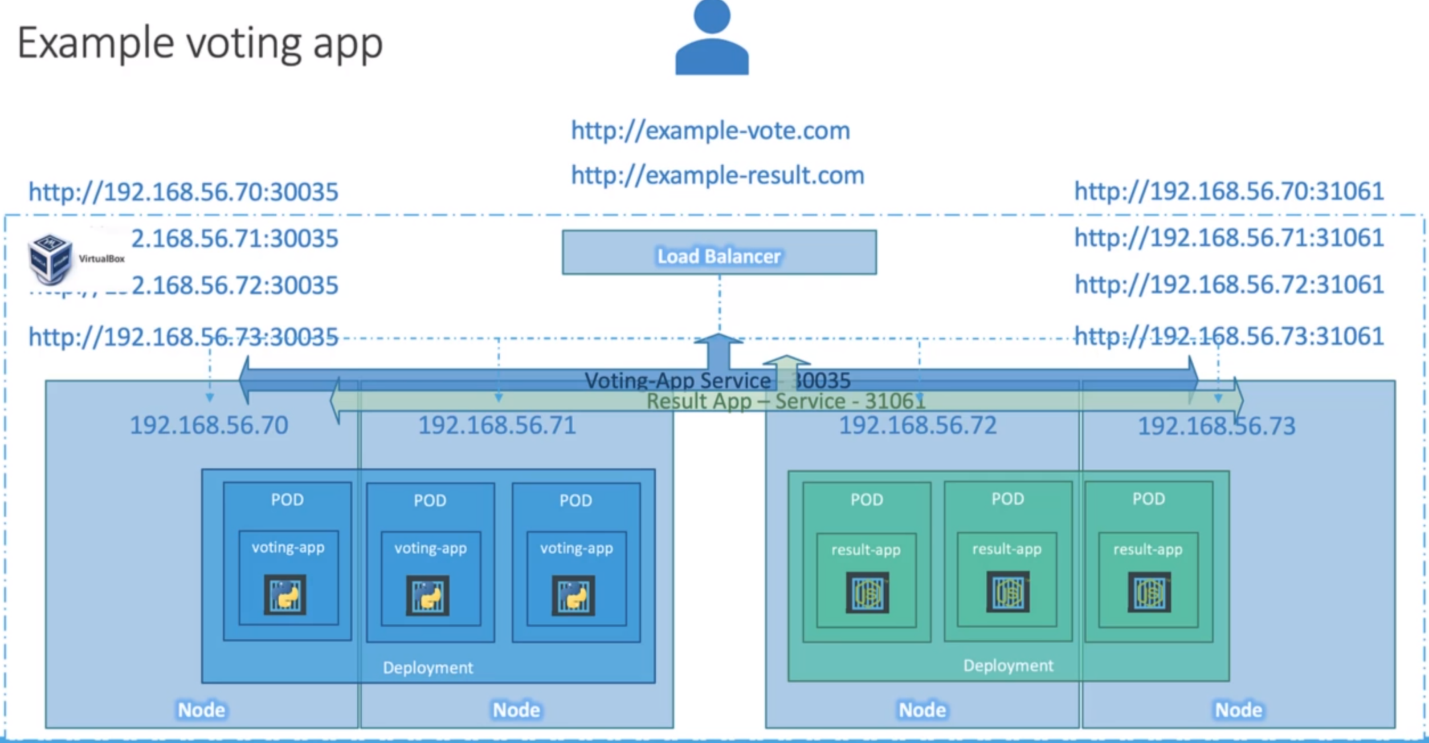
The services with type NodePort help in receiving traffic on the ports, on the nodes and routing the traffic to the respective pods. But what URL would you give your end users to access the applications?

You could access any of these two applications using IP of any of the nodes and high port with the service is exposed on. So that would be four IP and Port combinations for the voting-app and four IP and Port combination for the result-app. Note that even if your ports are only hosted on two of the nodes, they will still be accessible on the IPs of all the nodes in the cluster, say the pods for the voting app are only deployed on the nodes with IP 70 and 71, they would still be accessible on the ports of all the nodes in the cluster. That’s how a service is configured.

So, you would share these URLs to your users to access the application, but that’s not what the end user wants. They need a single URL like <http://example-vote.com> or <http://example-result.com> to access the application. How do you achieve that?

One way to achieve this is to create a new VM for load balancer purpose and install and configure a suitable load balancer on it like proxy. Then configure the load balancer to route traffic to the underlying nodes. Setting all of that external load balancing and then maintaining and managing that can be a tedious task. However, if we are on a supported cloud platform like Google Cloud, AWS or Azure, we could leverage the native load balancer of that cloud platform.

Kubernetes has support for integrating with the native load balancers of certain cloud providers and configuring that for us. All you need to do is set the service type for the front-end services to load balancer instead of NodePort. Remember, this only works with supported cloud platforms.



In an unsupportive environment like VirtualBox or any other environments, then it would have the same effect as setting it to NodePort, where the services are exposed on a high-end port on the nodes.