SSH to the **haproxyVM on Azure or any of the master NODE** of the 3 AWS Clusters

1. Install the mandatory deployment

|  |
| --- |
| $ sudo su  # kubectl apply -f https://raw.githubusercontent.com/kubernetes/ingress-nginx/master/deploy/mandatory.yaml |

2. Install the Nginx-Ingress controller

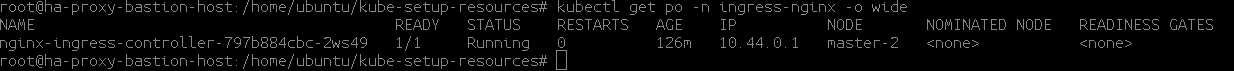
|  |
| --- |
| # kubectl apply -f https://raw.githubusercontent.com/kubernetes/ingress-nginx/master/deploy/provider/baremetal/service-nodeport.yaml |

3. Once, the Ingress is deployed.

Run the below command to check the ingress **NODE where the ingress controller is running**

|  |
| --- |
| # kubectl get po -n ingress-nginx -o wide |

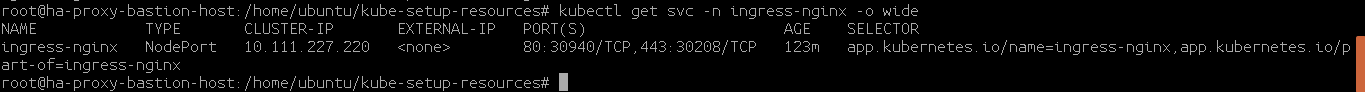
Example



Run the below command to check the ingress **NODEPORT of port 80 and port 443 on which the ingress controller is exposed.**

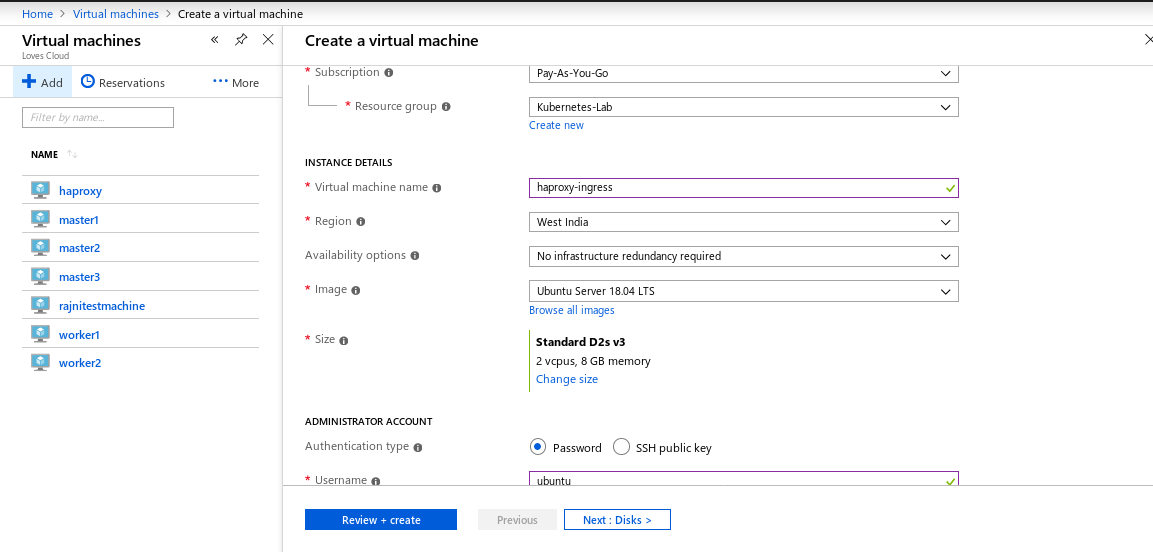
|  |
| --- |
| # kubectl get svc -n ingress-nginx -o wide |

Example



Note down the hostname Private-IP of the NODE where the Ingress controller is deployed from the above command, also note down the http port i.e binded to port 80 and https port i.e binded to port 443.

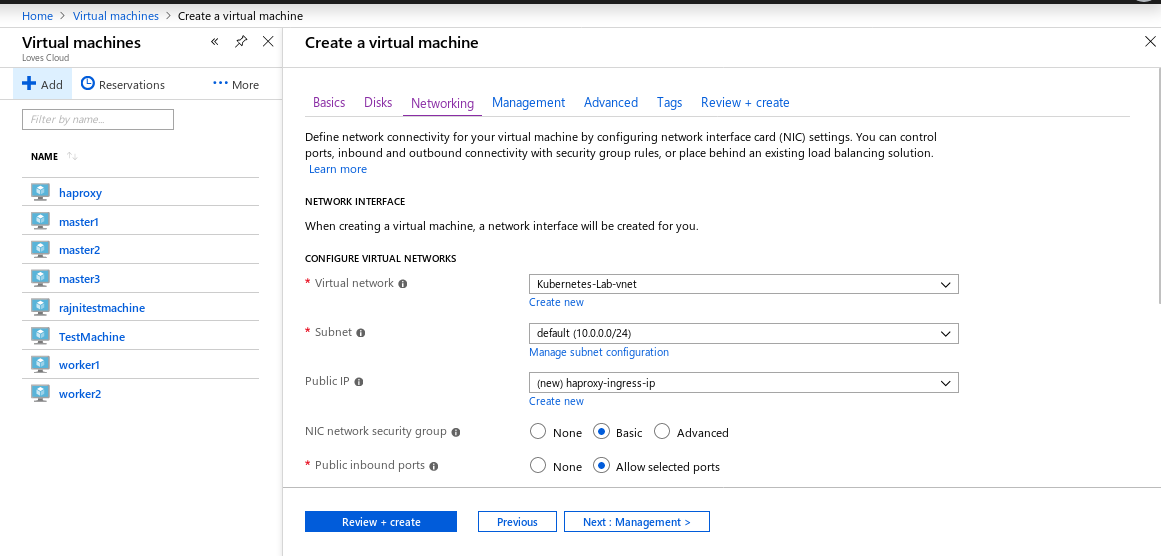
4. Launch an ubuntu server 18.04 lts VM of Standard B2s on Azure or **t3.medium** if using **AWS** in **DevOps Security Group** with Virtual Machine name as “**haproxy-ingress**” in the same resource group as of your cluster and username as “**ubuntu**” on your Azure account.



**On Azure do the below settings.**

**Open PORT 443,80 and 22 while creating the haproxy-ingress VM**

**Select the Virtual Network same as of your cluster resource group**

****

**5. SSH to the “haproxy-ingress” VM/Instance on a different terminal and RUN the below commands.**

|  |
| --- |
| $ sudo su # apt-get update # sudo apt-get -y install haproxy # cd /etc/haproxy/ # mv haproxy.cfg haproxy.cfg.backup  # vim haproxy.cfg |

**6. Paste the below script to the haproxy.cfg**

Now, update the HA-Proxy configuration by adding the Private IP and HOSTNAME of the **NODE** where the Ingress Controller is deployed and the ports of the ingress controller.

Please update the below fields in the script below.

**HOSTNAME** : hostname of the NODE where the Ingress controller is deployed, **NODE-internal-ip** : check the NODE’s internal-IP from Azure/AWS portal on which the Ingress controller is deployed.

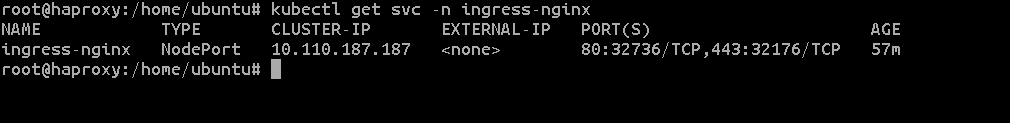
**haproxy-ingress-internal-ip :** check the Haproxy-ingress-internal-ip from Azure/AWS portal.

|  |
| --- |
| global  user haproxy  group haproxy  defaults  mode http  log global  retries 2  timeout connect 3000ms  timeout server 5000ms  timeout client 5000ms  frontend Ingress-http  bind <haproxy-ingress-internal-ip>:80  option tcplog  mode tcp  default\_backend Ingress-nginx-http  backend Ingress-nginx-http  mode tcp  balance roundrobin  option tcp-check  server **HOSTNAME** <NODE-internal-ip>:<http-port> check fall 3 rise 2  frontend Ingress-https  bind <haproxy-ingress-internal-ip>:443  option tcplog  mode tcp  default\_backend Ingress-nginx-https  backend Ingress-nginx-https  mode tcp  balance roundrobin  option tcp-check  server HOSTNAME <NODE-internal-ip>:<https-port> check fall 3 rise 2 |

**Example**

In the below example the **nginx-ingress controller** is deployed on the **worker1(**  hence HOSTNAME is worker1 **)**, the **haproxy-ingress-internal-ip** is **10.0.0.10**  **(** *where 10.0.0.10 is Internal IP of haproxy-ingress taken from portal***)**and the **nginx-ingress** is bind on **10.0.0.8:32736 (** *where 10.0.0.8 is Internal IP of worker 1 taken from portal***)** for **http** and **10.0.0.8:32176** for **https**

****



|  |
| --- |
| global  user haproxy  group haproxy defaults  mode http  log global  retries 2  timeout connect 3000ms  timeout server 5000ms  timeout client 5000ms frontend Ingress-http  bind 10.0.0.10:80  option tcplog  mode tcp  default\_backend Ingress-nginx-http backend Ingress-nginx-http  mode tcp  balance roundrobin  option tcp-check  server worker1 10.0.0.8:32736 check fall 3 rise 2  frontend Ingress-https  bind 10.0.0.10:443  option tcplog  mode tcp  default\_backend Ingress-nginx-https backend Ingress-nginx-https  mode tcp  balance roundrobin  option tcp-check  server worker1 10.0.0.8:32176 check fall 3 rise 2 |

**7.** Restart the **Haproxy** service

|  |
| --- |
| # systemctl restart haproxy # systemctl status haproxy |

**Nginx-Ingress has been successfully deployed.**

Here, Nginx-Ingress has been configured and running behind the haproxy-ingress VM. It can be accessed from the public-ip of the haproxy-ingress controller followed by the path of the ingress resources that is explained in lab-7.