.

Lesson 05 Demo 04

Publishing Swarm Service Ports

Objective: To publish a swarm service's port to external hosts to enable access to the service from outside the Docker swarm network

Tools required: Ubuntu OS and Docker

Prerequisites: None

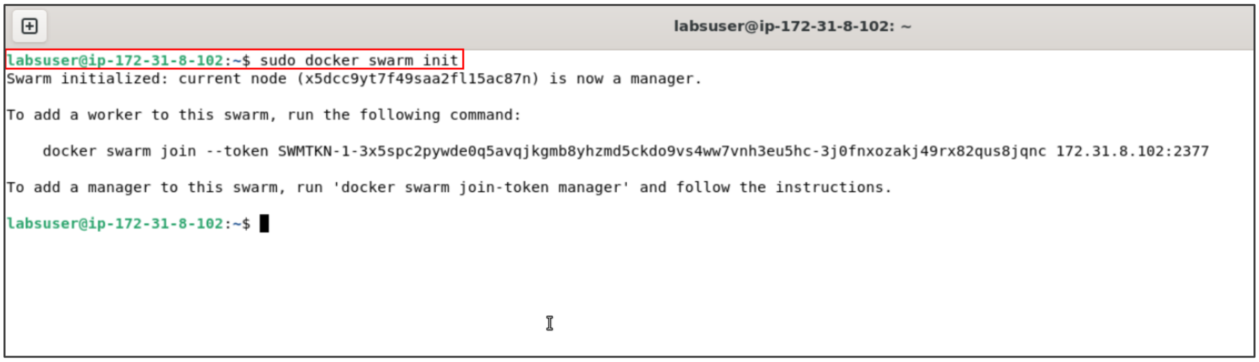
Steps to be followed:

1. Publish swarm service port for external access

Step 1: Publish swarm service port for external access

1.1 Initialize the Docker swarm using the following command:

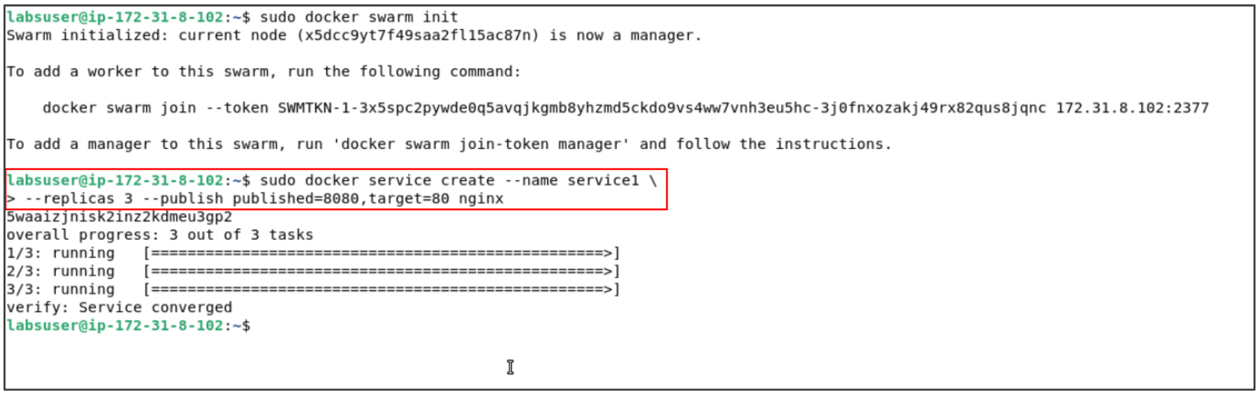
sudo docker swarm init



```
labsuser@ip-172-31-8-102:~$ sudo docker swarm init
Swarm initialized: current node (x5dcc9yt7f49saa2fl15ac87n) is now a manager.

To add a worker to this swarm, run the following command:

    docker swarm join --token SWMTKN-1-3x5spc2pywde0q5avqjkgmb8yhzmd5ckdo9vs4ww7vnh3eu5hc-3j0fnxozakj49rx82qus8jqnc 172.31.8.102:2377

To add a manager to this swarm, run 'docker swarm join-token manager' and follow the instructions.

labsuser@ip-172-31-8-102:~$ 
```

.

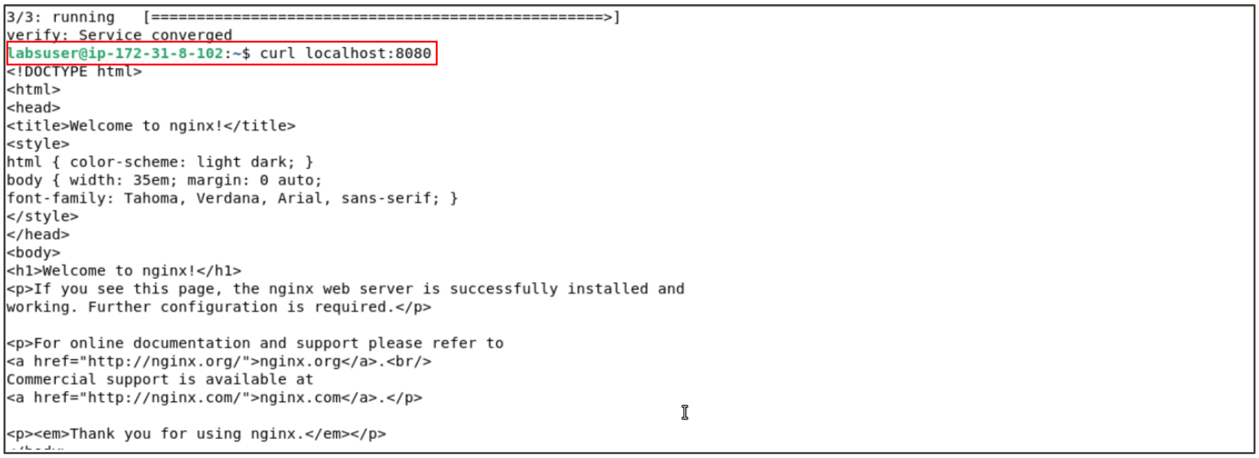
1.2 Execute the following commands to publish a swarm service’s port using the Routing Mesh:
**sudo docker service create --name service1 **
--replicas 3 --publish published=8080,target=80 nginx



```
labsuser@ip-172-31-8-102:~$ sudo docker swarm init
Swarm initialized: current node (x5dcc9yt7f49saa2fl15ac87n) is now a manager.

To add a worker to this swarm, run the following command:

    docker swarm join --token SWMTKN-1-3x5spc2pywde0q5avqjkgmb8yhzmd5ckdo9vs4ww7vnh3eu5hc-3j0fnxozakj49rx82qus8jqnc 172.31.8.102:2377

To add a manager to this swarm, run 'docker swarm join-token manager' and follow the instructions.

labsuser@ip-172-31-8-102:~$ sudo docker service create --name service1 \
> --replicas 3 --publish published=8080,target=80 nginx
5waaizjnisk2inz2kdmeu3gp2
overall progress: 3 out of 3 tasks
1/3: running   [==================================================>]
2/3: running   [==================================================>]
3/3: running   [==================================================>]
verify: Service converged
labsuser@ip-172-31-8-102:~$
```

Note: You can also write the above command in the following way: **docker service create --name service1 --replicas 3 -p 8080:80 nginx**

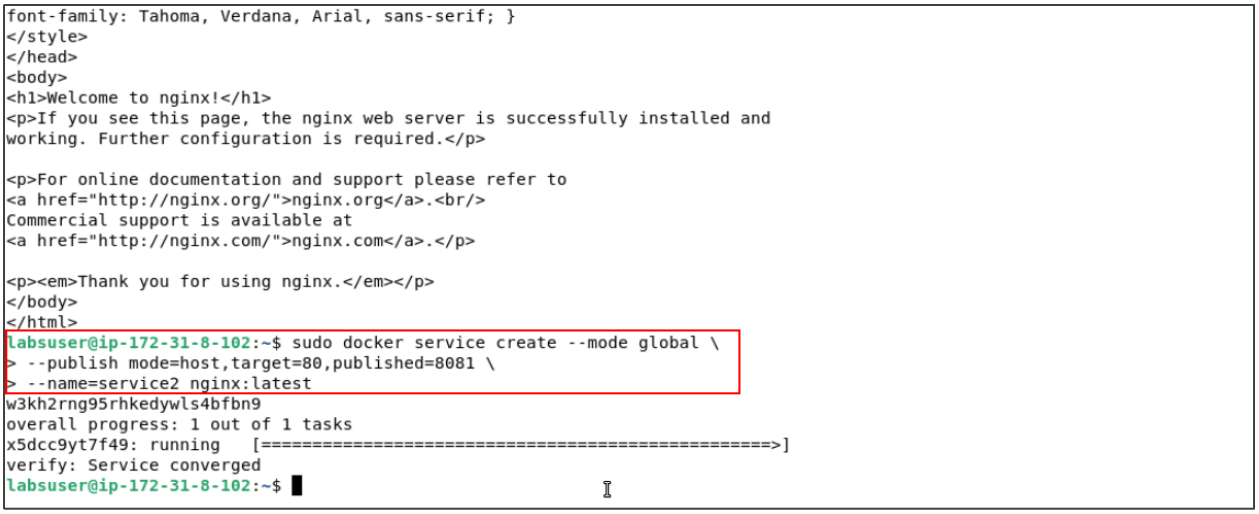
1.3 Use the following command to check whether your service has started on port 8080:
curl localhost:8080



```
3/3: running   [==================================================>]
verify: Service converged
labsuser@ip-172-31-8-102:~$ curl localhost:8080
<!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>
```

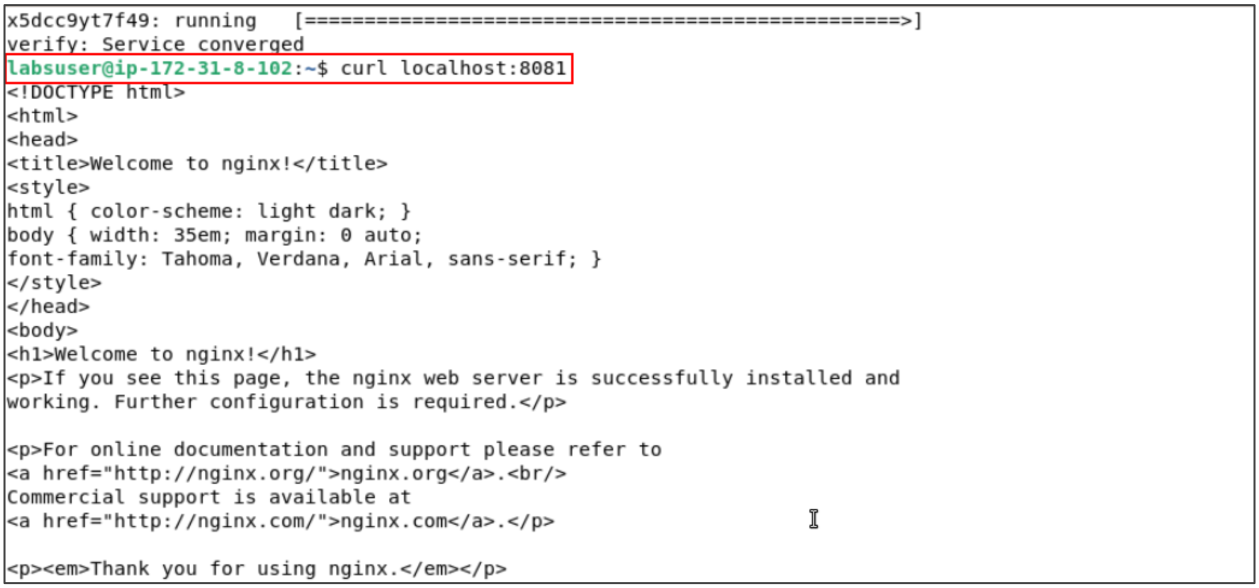
.

1.4 Use the **mode=host** option with the **--publish** flag along with the **--mode** global flag to publish a port directly on the swarm node, as shown in the screenshot below:
sudo docker service create --mode global \
--publish mode=host,target=80,published=8081 \
--name=service2 nginx:latest



```
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>
labsuser@ip-172-31-8-102:~$ sudo docker service create --mode global \
> --publish mode=host,target=80,published=8081 \
> --name=service2 nginx:latest
w3kh2rng95rhkedywls4bfbn9
overall progress: 1 out of 1 tasks
x5dcc9yt7f49: running   [==================================================>]
verify: Service converged
labsuser@ip-172-31-8-102:~$
```

1.5 Use the following command to check whether your service has started on port 8081:
curl localhost:8081



```
x5dcc9yt7f49: running   [==================================================>]
verify: Service converged
labsuser@ip-172-31-8-102:~$ curl localhost:8081
<!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>
```

By following these steps, you have successfully published a swarm service's port to external hosts, enabling access to the service from outside the Docker swarm network.