**Lesson 05 Demo 03**

**Creating a Host Network**

**Objective:** To create a host network for a standalone container and enable direct binding to the Docker host's network interface for seamless integration and optimal network performance

**Tools required:** Ubuntu OS and Docker

**Prerequisites:** None

Steps to be followed:

1. Initialize Docker swarm and create a standalone container
2. Inspect and verify container networking

**Step 1: Initialize Docker swarm and create a standalone container**

1. Run the following command to create and start a container as a detached process using the host networking driver: **sudo docker run --rm -d --network host --name nginx\_container1 nginx**

A screen shot of a computer

Description automatically generated

|  |
| --- |
| **Note:** The host networking driver works only on Linux hosts and is not supported on Docker Desktop for Mac or Windows Server. |

1. Open the browser and navigate to **http://localhost:80/**   
     
   A screenshot of a computer

   Description automatically generated

**Step 2: Inspect and verify container networking**

1. Inspect the container to check the **NetworkMode** under the **HostConfig** using the following command:  
   **sudo docker container inspect nginx\_container1**  
     
   A screenshot of a computer

   Description automatically generated

A screenshot of a computer

Description automatically generated

1. Verify the process that is bound to port 80 using the following netstat command:  
   **sudo netstat -tulpn | grep :80**  
     
   A screenshot of a computer

   Description automatically generated
2. Examine all the network interfaces using the following command:  
   **ip addr show**  
     
   *A screenshot of a computer

   Description automatically generated*
3. Stop the container using the following command:  
   **sudo docker container stop nginx\_container1**  
   A screenshot of a computer

   Description automatically generated  
     
   By following these steps, you have successfully created a host network for a standalone container to facilitate direct binding to the Docker host's network interface and ensure seamless integration and optimal network performance.