

## SOFTWARE ENGINEERING LAB

### EXERCISE – 7

### TOPIC – 6

## DEPLOYING AND MANAGING MONITORING SYSTEMS USING NAGIOS IN DOCKER

**In this exercise, we will be:**

- Setting up and running Nagios using Docker for quick and simple installation.
- Accessing the Nagios Dashboard to explore its features.
- Monitoring the health and status of systems and services effectively.
- Learning how to manage checks, notifications, and downtime settings.
- Understanding how to stop and remove the Nagios Docker container when finished.

• **Note: At every step take screenshots and save in a document**

### Step 1: Pulling the Nagios Image

#### 1. Command to Pull Nagios

Open a terminal and type this command to download the Nagios image:

```
docker pull jasonrivers/nagios:latest
```

- This command tells Docker to download the **latest version** of Nagios from a specific repository (**jasonrivers**).

### Step 2: Running Nagios

#### 1. Command to Run Nagios

```
docker run --name nagiosdemo -p 8888:80 jasonrivers/nagios:latest
```

**Explanation of Each Part:**

- **--name nagiosdemo**: Names the container **nagiosdemo** so you can easily identify it later.
- **-p 8888:80**: Maps your computer's **port 8888** to the container's **port 80** (port 80 is used for web access).
- **jasonrivers/nagios:latest**: Specifies the software image (Nagios) and its version.

### Step 3: Accessing Nagios

1. Open a browser and type:

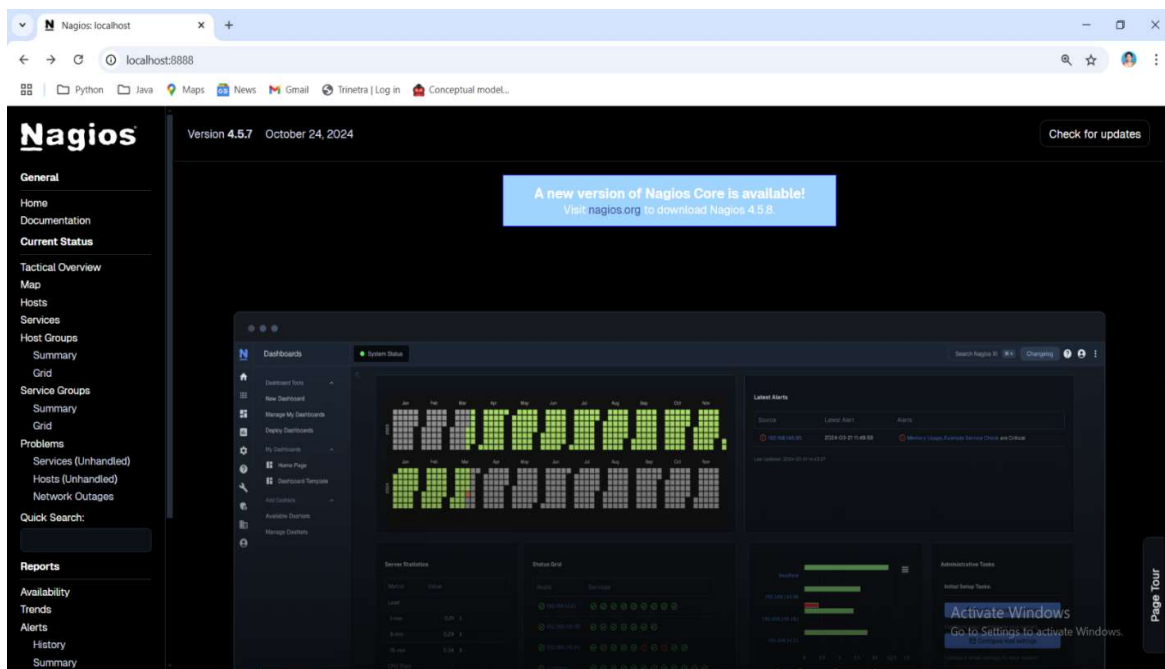
**localhost:8888**

This will open the Nagios web interface.

2. Login Credentials:

- Username: **nagiosadmin**
- Password: **nagios**

### Understanding the Nagios Dashboard



#### 1. General Information

- **Nagios Version:** The top-left corner shows the version of Nagios you're using (e.g., Version 4.5.7).

## 2. Navigation Panel (Left-Side Menu)

- **Home:** Returns you to the main dashboard.
- **Hosts:** Displays the list of systems (hosts) being monitored.
- **Services:** Shows activities or tasks being monitored, such as CPU usage, disk space, or network status.
- **Reports:** Provides historical data and trends about your systems.

## 3. Dashboard Content

### a. System Status (Top Section)

- **Green Boxes:** Indicate periods when everything was working fine.
- **Red or Yellow Boxes:** Indicate problems or warnings.

### b. Latest Alerts

- **Source:** Shows which system (e.g., IP address) had the issue.
- **Alert:** Describes the problem (e.g., "Memory usage critical").

### c. Server Statistics

- **Load:** How much work the server is handling.
- **CPU Stats:** How busy the processor is.

### d. Status Grid

- **Green Circles:** Everything is working fine.
- **Red or Yellow Circles:** Indicate issues.

## 4. Monitoring Specific Hosts and Services

### 1. Monitor Hosts:

- Go to the **Hosts** menu on the left.
- You'll see a list of systems being monitored.

## 2. Monitor Services:

- Click on a host to see details about the services being monitored, such as CPU usage or memory status.

The screenshot shows the Nagios web interface for the host 'localhost'. The left sidebar contains navigation links for General, Current Status, Tactical Overview, Map, Hosts, Services, Host Groups, Service Groups, Problems, and Reports. The main content area is divided into several sections:

- Host Information:**
  - Last Updated: Sun Dec 8 13:24:40 UTC 2024
  - Updated every 90 seconds
  - Nagios® Core™ 4.5.7 - www.nagios.org
  - Logged in as nagiosadmin
  - View Status Detail For This Host
  - View Alert History For This Host
  - View Trends For This Host
  - View Alert Histogram For This Host
  - View Availability Report For This Host
  - View Notifications For This Host
- Host State Information:**
  - Host Status:** UP (for 0d 0h 12m 32s)
  - Status Information:** PING OK - Packet loss = 0%, RTA = 0.03 ms
  - Performance Data:** rta=0.034000ms;3000.000000;5000.000000;0.000000 pl=0%;80;100;0
  - Current Attempt:** 1/10 (HARD state)
  - Last Check Time:** 12-08-2024 13:24:14
  - Check Type:** ACTIVE
  - Check Latency / Duration:** 0.000 / 4.200 seconds
  - Next Scheduled Active Check:** 12-08-2024 13:29:14
  - Last State Change:** 12-08-2024 13:12:08
  - Last Notification:** N/A (notification 0)
  - Is This Host Flapping?** NO (0.00% state change)
  - In Scheduled Downtime?** NO
  - Last Update:** 12-08-2024 13:24:36 ( 0d 0h 0m 4s ago)
  - Active Checks:** ENABLED
  - Passive Checks:** ENABLED
  - Obsessing:** ENABLED
  - Notifications:** ENABLED
  - Event Handler:** ENABLED
  - Flap Detection:** ENABLED
- Host Commands:**
  - Locate host on map
  - Disable active checks of this host
  - Re-schedule the next check of this host
  - Submit passive check result for this host
  - Stop accepting passive checks for this host
  - Stop obsessing over this host
  - Disable notifications for this host
  - Send custom host notification
  - Schedule downtime for this host
  - Schedule downtime for all services on this host
  - Disable notifications for all services on this host
  - Enable notifications for all services on this host
  - Schedule a check of all services on this host
  - Disable checks of all services on this host
  - Enable checks of all services on this host
  - Disable event handler for this host
  - Disable flap detection for this host
  - Clear flapping state for this host
- Host Comments:**
  - Add a new comment
  - Delete all comments

The bottom of the page shows a table for Host Comments with columns: Entry Time, Author, Comment, Comment ID, Persistent, Type, Expires, and Actions.

## Step 4: Exploring the Host Information Page

### a. Top Section (Basic Host Details)

- **Host:** The system being monitored is named **localhost** (your own computer).
- **Member of:** This host belongs to the **linux-servers** group.
- **IP Address:** The host's IP address is **127.0.0.1** (this means it's your local system).

### b. Middle Section (Host State Information)

This section shows the **current health and performance** of the host:

- **Host Status:** UP (green color) means the system is working fine.
- **Status Information:** Nagios checks if the system is alive using a "ping" command.
  - **No Packet Loss:** The system is responding properly.

- **Round-Trip Time (RTA):** The time it takes for a message to go to the system and back is very fast.
- **Last Check Time:** The last time Nagios checked the system.
- **Next Check:** When the next check will happen.
- **Flapping:** Flapping occurs when a system repeatedly goes up and down. Here, it's stable (NO).
- **Active Checks:** Monitoring features like active checks and notifications are all **ENABLED** (green boxes).

### c. Right Section (Host Commands)

You can perform actions for this host using these commands:

- **Locate Host on Map:** See the host's location on the network map.
- **Disable Active Checks:** Stop Nagios from checking this host automatically.
- **Re-schedule Next Check:** Force Nagios to check the host immediately.
- **Disable Notifications:** Turn off alerts for this host.
- **Schedule Downtime:** If you plan to shut down the host, use this to avoid unnecessary alerts.

## Stopping Nagios

To stop Nagios, we need to stop the Docker container running it.

1. **Find the Container Name:** If you used the earlier command to start Nagios, the container name is **nagiosdemo**. If you're unsure about the name, follow these steps:
  - Open your terminal.
  - Type this command to see all running containers:
2. **Stop the Container:** Use the following command to stop the Nagios container:

```
docker ps
```

- You'll see a list of containers. Look for the **CONTAINER NAME** column to find the container running Nagios (e.g., **nagiosdemo**).

```
docker stop nagiosdemo
```

- **nagiosdemo**: This is the name of the container. Replace it with the actual container name if it's different.

## Step 2: Deleting the Nagios Container

If you no longer need Nagios, you can delete the container.

1. **Remove the Container:** Use this command to delete the container:

```
docker rm nagiosdemo
```

- This permanently deletes the Nagios container.
- If the container is still running, you'll see an error. In that case, ensure you've stopped it first (Step 1).

## Step 3: Deleting the Nagios Image

1. **Find the Image Name:** To see all Docker images on your system, type:

```
docker images
```

- Look for the **IMAGE NAME** column. You should see something like **jasonrivers/nagios**.

2. **Delete the Image:** Use this command to remove the image:

```
docker rmi jasonrivers/nagios:latest
```

- If the image is still being used by a container, you'll get an error. Make sure you've already deleted the container (Step 2).