

A.Bhavagna

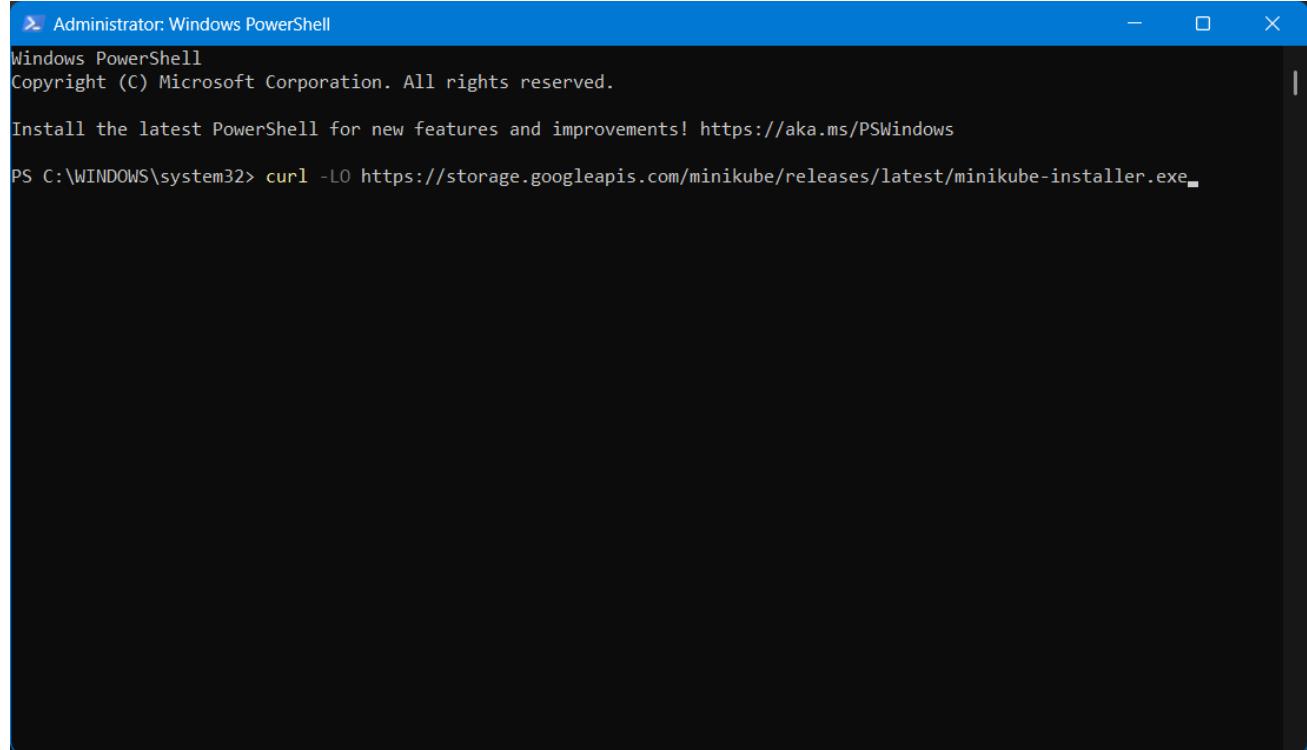
23BD1A050C

Working with minikube and Nagios

Minikube Installation :

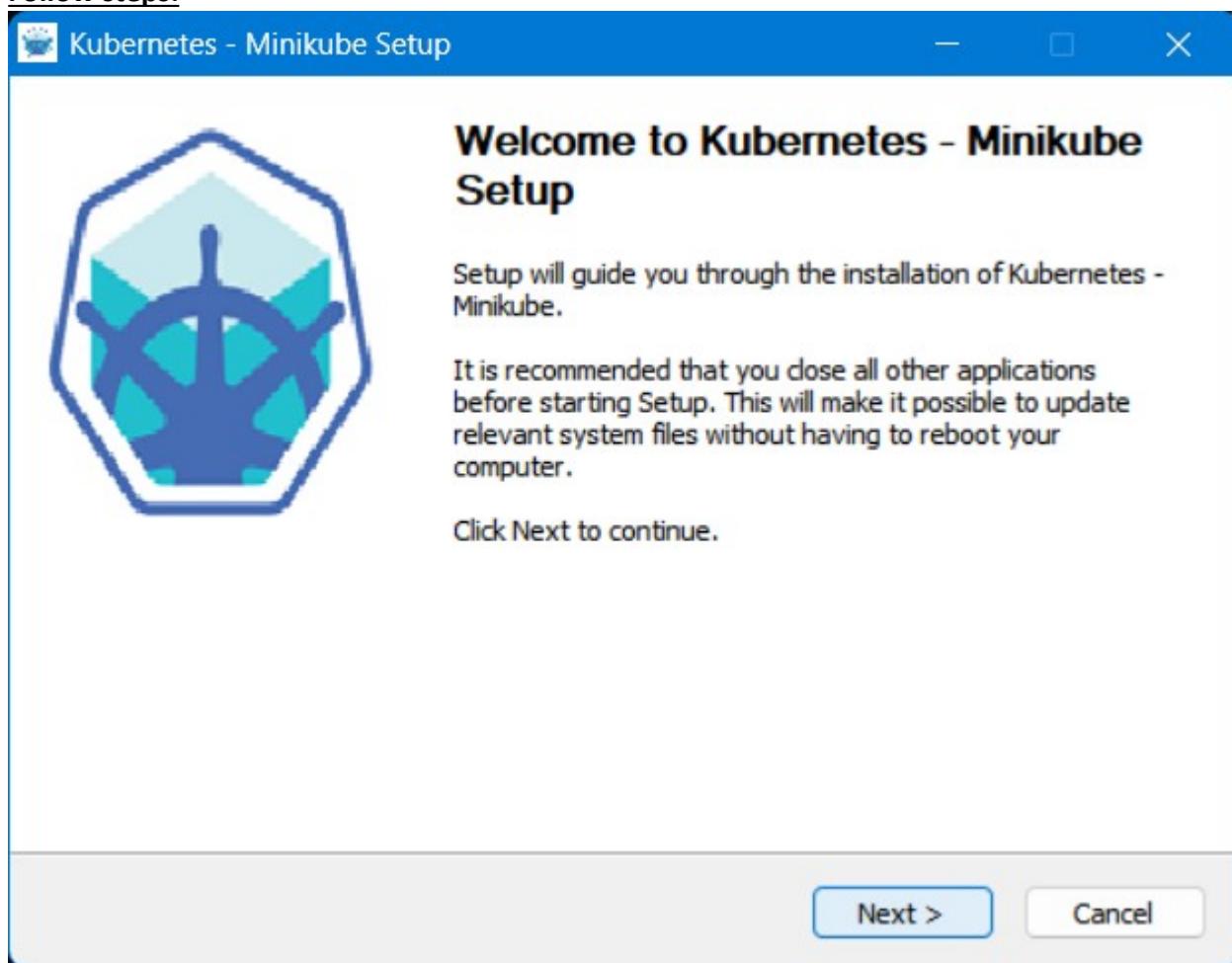
Run command in powershell

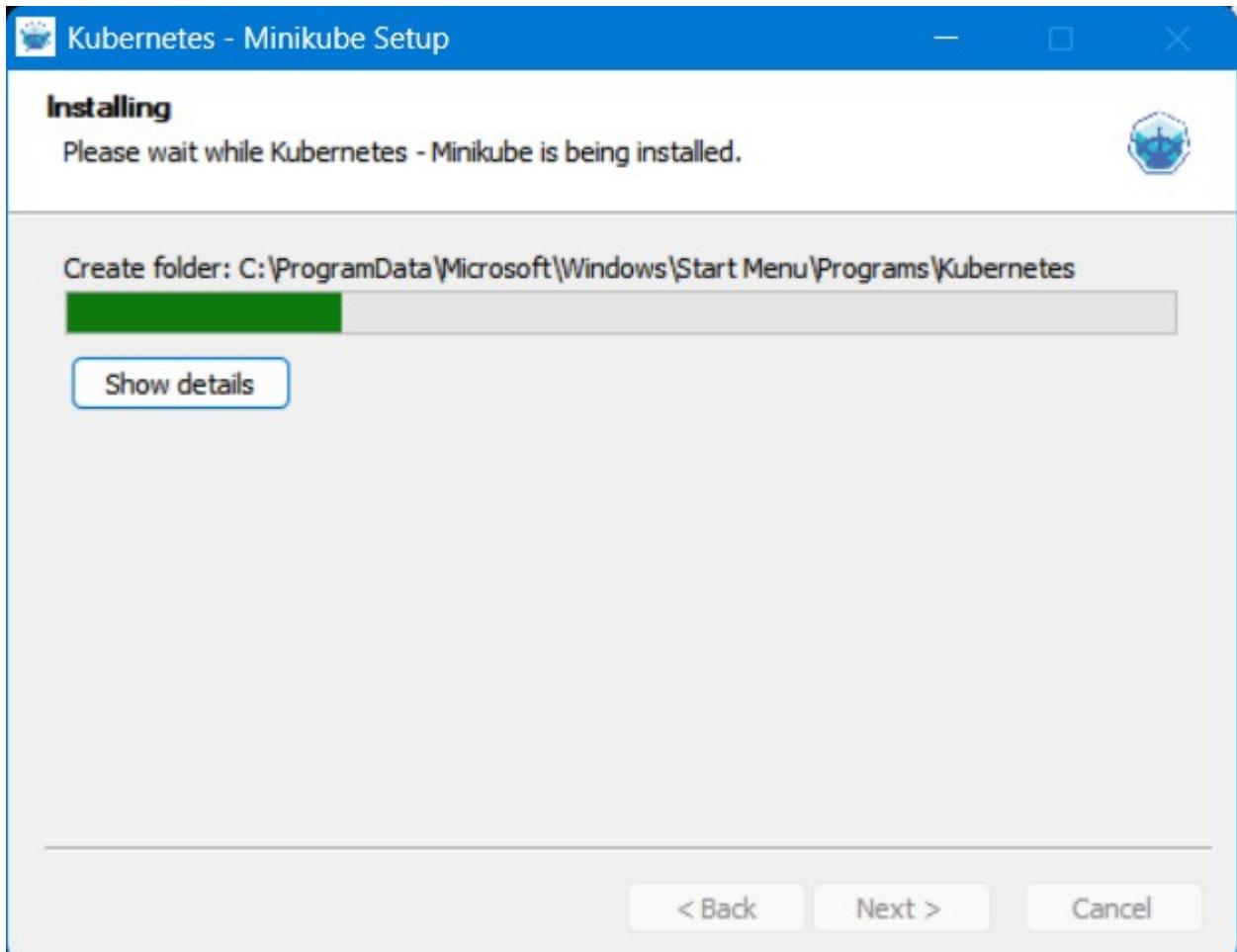
```
curl -LO https://storage.googleapis.com/minikube/releases/latest/minikube-linux-amd64
```



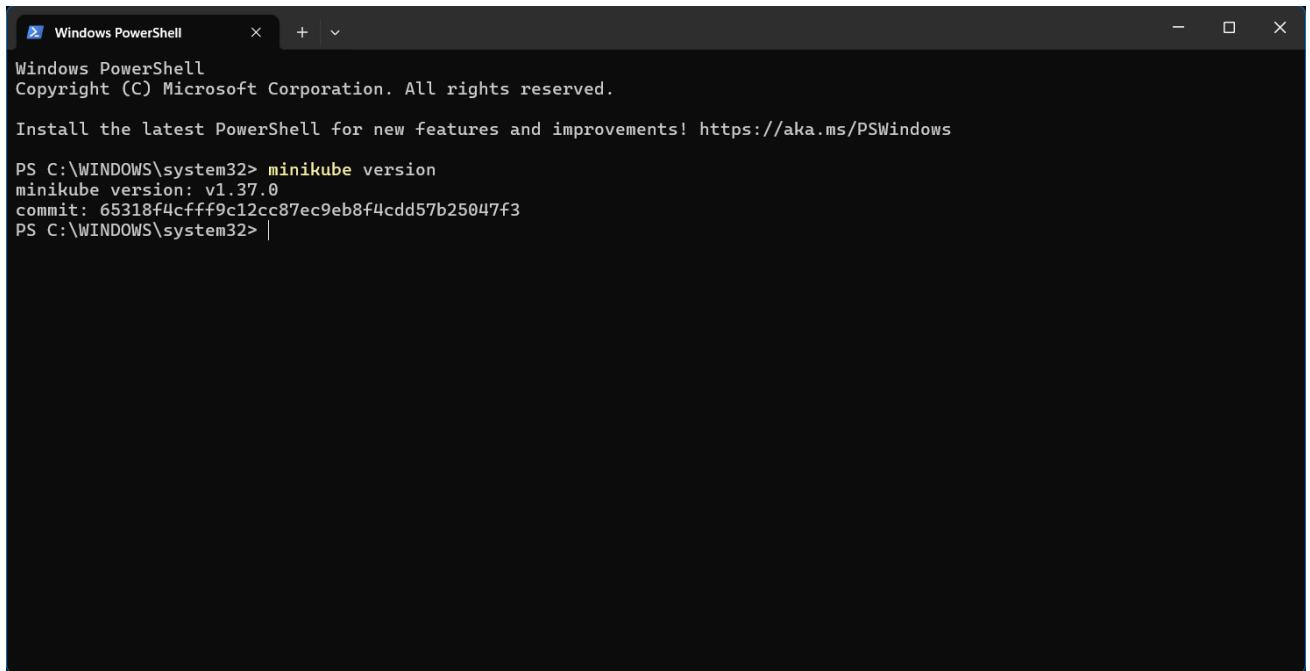
The screenshot shows a Windows PowerShell window titled "Administrator: Windows PowerShell". The title bar is blue with white text. The main area of the window is black with white text. At the top, it says "Windows PowerShell" and "Copyright (C) Microsoft Corporation. All rights reserved.". Below that, there is a message: "Install the latest PowerShell for new features and improvements! <https://aka.ms/PSWindows>". At the bottom, the command "curl -LO https://storage.googleapis.com/minikube/releases/latest/minikube-installer.exe" is typed into the command line.

Follow steps:





Verify Installation:



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

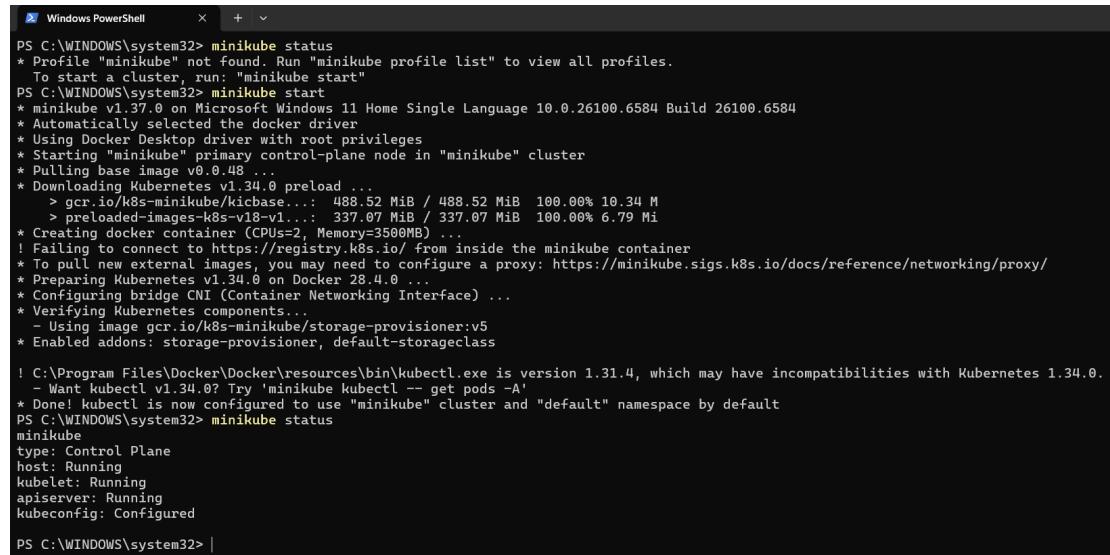
PS C:\WINDOWS\system32> minikube version
minikube version: v1.37.0
commit: 65318f4cff9c12cc87ec9eb8f4cdd57b25047f3
PS C:\WINDOWS\system32> |
```

Minikube Automation Steps

Step 1: Start Minikube Cluster

- Open your terminal and run the command:

```
minikube start
```



```
Windows PowerShell
PS C:\WINDOWS\system32> minikube status
* Profile "minikube" not found. Run "minikube profile list" to view all profiles.
  To start a cluster, run: "minikube start"
PS C:\WINDOWS\system32> minikube start
* minikube v1.37.0 on Microsoft Windows 11 Home Single Language 10.0.26100.6584 Build 26100.6584
* Automatically selected the docker driver
* Using Docker Desktop driver with root privileges
* Starting "minikube" primary control-plane node in "minikube" cluster
* Pulling base image v0.0.48 ...
* Downloading Kubernetes v1.34.0 preload ...
  > gcr.io/k8s-minikube/kicbase...: 488.52 MiB / 488.52 MiB 100.00% 10.34 M
  > preloaded-images-k8s-v18-v1...: 337.07 MiB / 337.07 MiB 100.00% 6.79 Mi
* Creating docker container (CPUs=2, Memory=3500MB) ...
! Failing to connect to https://registry.k8s.io/ from inside the minikube container
* To pull new external images, you may need to configure a proxy: https://minikube.sigs.k8s.io/docs/reference/networking/proxy/
* Preparing Kubernetes v1.34.0 on Docker 28.4.0 ...
* Configuring bridge CNI (Container Networking Interface) ...
* Verifying Kubernetes components...
  - Using image gcr.io/k8s-minikube/storage-provisioner:v5
* Enabled addons: storage-provisioner, default-storageclass

! C:\Program Files\Docker\Dockers\resources\bin\kubectl.exe is version 1.31.4, which may have incompatibilities with Kubernetes 1.34.0.
  - Want kubectl v1.34.0? Try 'minikube kubectl -- get pods -A'
* Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
PS C:\WINDOWS\system32> minikube status
minikube
type: Control Plane
host: Running
kublet: Running
apiserver: Running
kubeconfig: Configured
PS C:\WINDOWS\system32> |
```

Step 2: Create and Manage Deployment

Create an application in Kubernetes:

```
kubectl create deployment mynginx --image=nginx
```

```
PS C:\WINDOWS\system32> kubectl create deployment mynginx --image=nginx
deployment.apps/mynginx created
PS C:\WINDOWS\system32> kubectl get deployment
NAME      READY   UP-TO-DATE   AVAILABLE   AGE
mynginx   0/1     1           0           14s
PS C:\WINDOWS\system32> kubectl get pods
NAME            READY   STATUS    RESTARTS   AGE
mynginx-645865c456-kkl4t  0/1   ContainerCreating   0          31s
PS C:\WINDOWS\system32> kubectl get deployment
NAME      READY   UP-TO-DATE   AVAILABLE   AGE
mynginx   1/1     1           1           2m30s
PS C:\WINDOWS\system32> kubectl get pods
NAME            READY   STATUS    RESTARTS   AGE
mynginx-645865c456-kkl4t  1/1   Running   0          2m33s
PS C:\WINDOWS\system32> |
```

Step 3: Scale the Deployment

```
PS C:\WINDOWS\system32> kubectl get pods
NAME            READY   STATUS    RESTARTS   AGE
mynginx-645865c456-kkl4t  1/1   Running   0          2m33s
PS C:\WINDOWS\system32> kubectl describe pod mynginx-645865c456-kkl4t
Name:           mynginx-645865c456-kkl4t
Namespace:      default
Priority:       0
Service Account: default
Node:          minikube/192.168.49.2
Start Time:    Tue, 14 Oct 2025 17:37:32 +0530
Labels:         app=mynginx
                pod-template-hash=645865c456
Annotations:   <none>
Status:        Running
IP:            10.244.0.3
IPs:
  IP:          10.244.0.3
Controlled By: ReplicaSet/mynginx-645865c456
Containers:
  nginx:
    Container ID: docker://0c0023d47fba3e9591852d158771cd46c0b6981c9ef6c2d481492738b75c98d2
    Image:          nginx
    Image ID:      docker-pullable://nginx@sha256:3b7732505933ca591ce4a6d860cb713ad96a3176b82f7979a8dfa9973486a0d6
    Port:          <none>
    Host Port:    <none>
    State:        Running
      Started:   Tue, 14 Oct 2025 17:38:03 +0530
    Ready:        True
    Restart Count: 0
    Environment:  <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-q9fcz (ro)
Conditions:
  Type        Status
  PodReadyToStartContainers  True
  Initialized  True
  Ready        True
  ContainersReady  True
  PodScheduled  True
Volumes:
  kube-api-access-q9fcz:
```

Step 4: Access the Nginx App

1. Using Port Forwarding:

```
kubectl port-forward svc/mynginx 8081:80
```

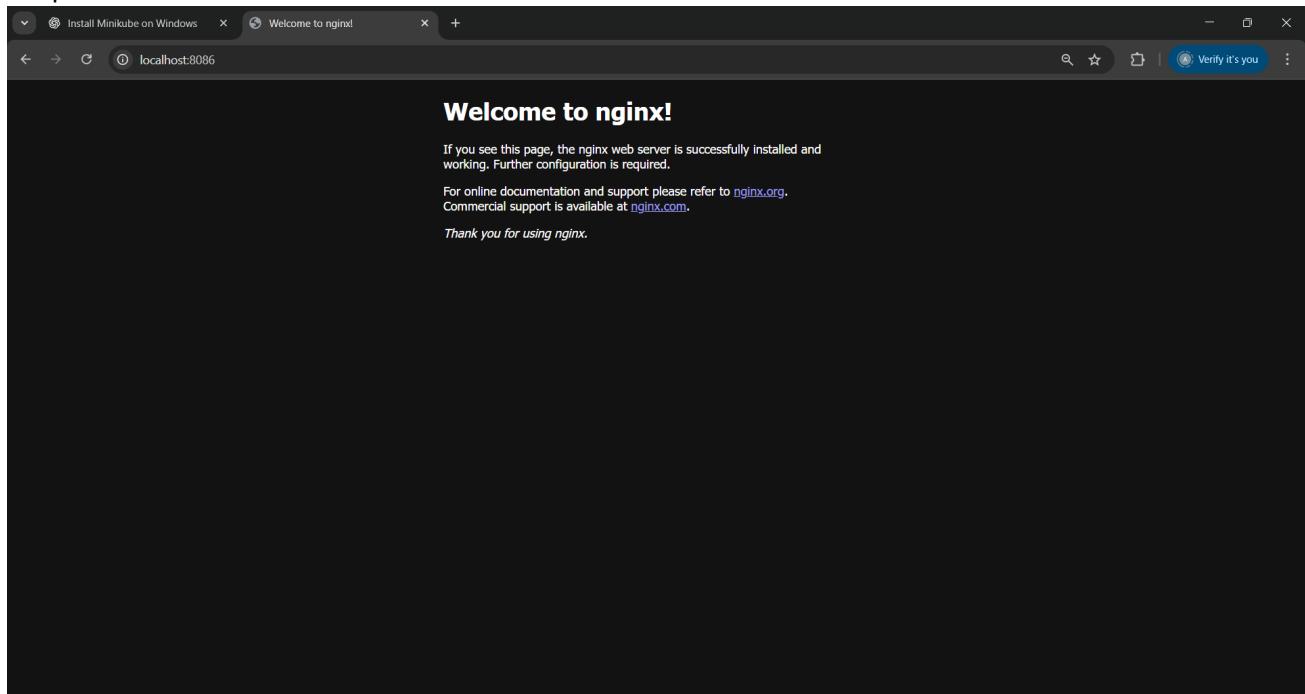
```

Windows PowerShell

Restart Count: 0
Environment: <none>
Mounts:
    /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-q9fcz (ro)
Conditions:
  Type        Status
  PodReadyToStartContainers True
  Initialized  True
  Ready       True
  ContainersReady True
  PodScheduled True
Volumes:
  kube-api-access-q9fcz:
    Type:      Projected (a volume that contains injected data from multiple sources)
    TokenExpirationSeconds: 3607
    ConfigMapName:          kube-root-ca.crt
    ConfigMapOptional:      <nil>
    DownwardAPI:           true
QoS Class:      BestEffort
Node-Selectors: <none>
Tolerations:    node.kubernetes.io/not-ready:NoExecute op=Exists for 300s
                node.kubernetes.io/unreachable:NoExecute op=Exists for 300s
Events:
  Type     Reason     Age   From            Message
  ----     ----     --   --              --
Normal  Scheduled  3m33s  default-scheduler  Successfully assigned default/mynginx-645865c456-kkl4t to minikube
Normal  Pulling    3m33s  kubelet         Pulling image "nginx"
Normal  Pulled    3m10s  kubelet         Successfully pulled image "nginx" in 22.351s (22.351s including waiting). Image size: 159974475 bytes.
Normal  Created   3m2s   kubelet         Created container: nginx
Normal  Started   3m2s   kubelet         Started container nginx
PS C:\WINDOWS\system32> kubectl expose deployment mynginx --type=NodePort --port=80 --target-port=80
service/mynginx exposed
PS C:\WINDOWS\system32> kubectl get service mynginx
Error from server (NotFound): services "mynginx" not found
PS C:\WINDOWS\system32> kubectl get service mynginx
NAME      TYPE      CLUSTER-IP      EXTERNAL-IP      PORT(S)      AGE
mynginx  NodePort  10.107.163.171  <none>          80:32030/TCP  30s
PS C:\WINDOWS\system32> kubectl port-forward svc/mynginx 8086:80
Forwarding from 127.0.0.1:8086 -> 80
Forwarding from [::1]:8086 -> 80
|

```

Output:

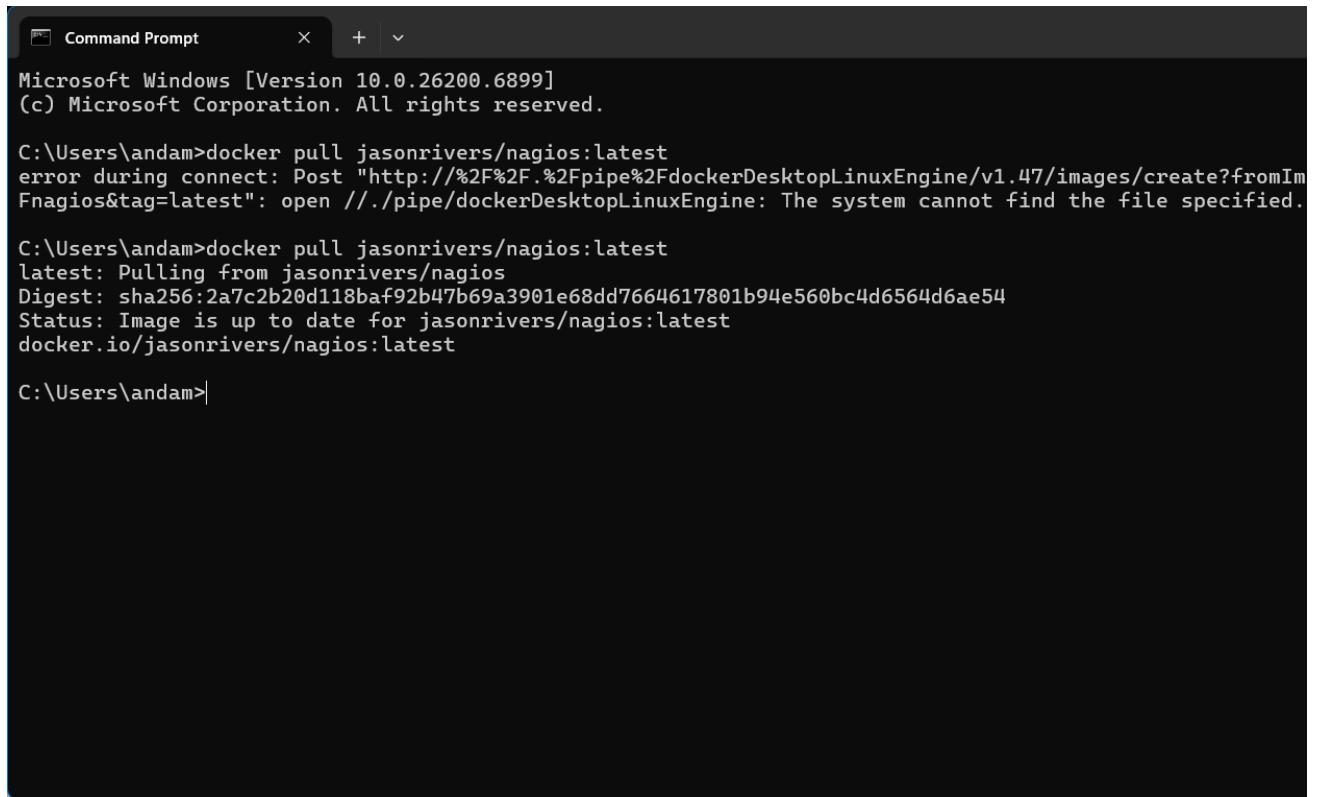


Nagios Automation Steps

Step 1: Pull the Nagios Docker Image

- Open a terminal and run:

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



```
Microsoft Windows [Version 10.0.26200.6899]
(c) Microsoft Corporation. All rights reserved.

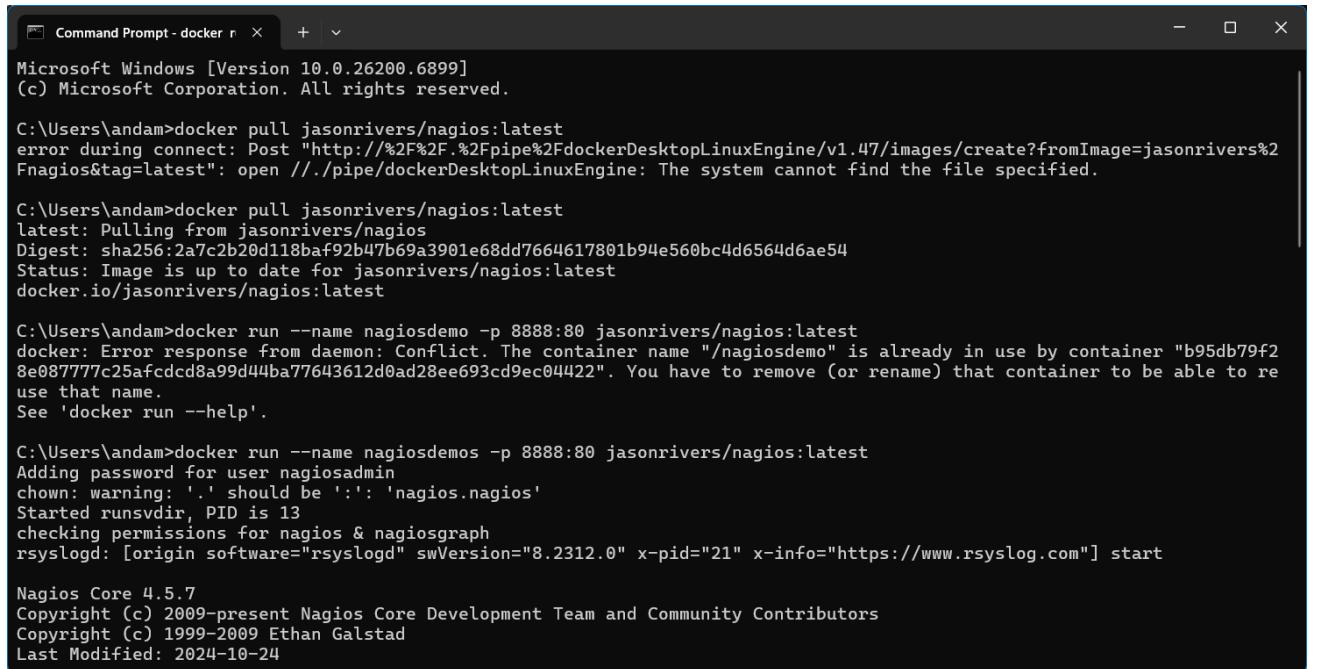
C:\Users\andam>docker pull jasonrivers/nagios:latest
error during connect: Post "http://%2F%2Fpipe%2FdockerDesktopLinuxEngine/v1.47/images/create?fromImage=jasonrivers%2Fnagios&tag=latest": open //./pipe/dockerDesktopLinuxEngine: The system cannot find the file specified.

C:\Users\andam>docker pull jasonrivers/nagios:latest
latest: Pulling from jasonrivers/nagios
Digest: sha256:2a7c2b20d118ba92b47b69a3901e68dd7664617801b94e560bc4d6564d6ae54
Status: Image is up to date for jasonrivers/nagios:latest
docker.io/jasonrivers/nagios:latest

C:\Users\andam>
```

Step 2: Run Nagios

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



```
Microsoft Windows [Version 10.0.26200.6899]
(c) Microsoft Corporation. All rights reserved.

C:\Users\andam>docker pull jasonrivers/nagios:latest
error during connect: Post "http://%2F%2Fpipe%2FdockerDesktopLinuxEngine/v1.47/images/create?fromImage=jasonrivers%2Fnagios&tag=latest": open //./pipe/dockerDesktopLinuxEngine: The system cannot find the file specified.

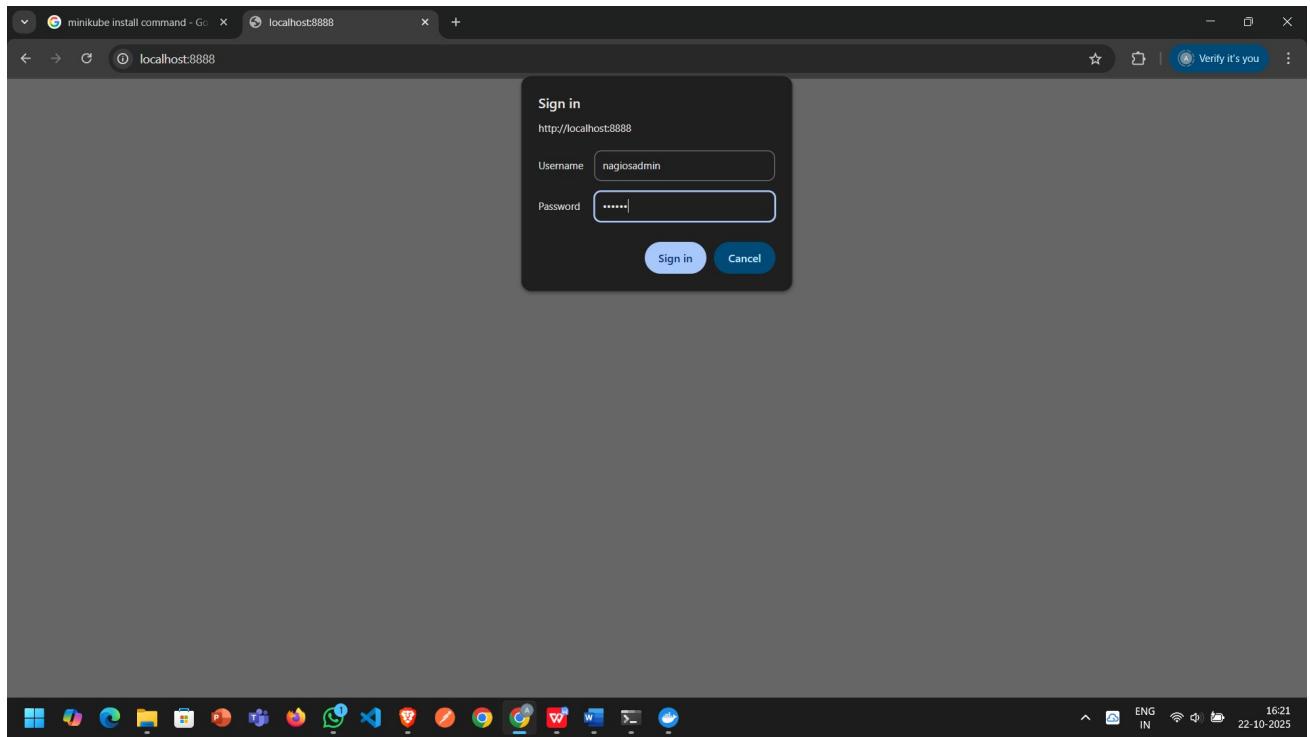
C:\Users\andam>docker pull jasonrivers/nagios:latest
latest: Pulling from jasonrivers/nagios
Digest: sha256:2a7c2b20d118ba92b47b69a3901e68dd7664617801b94e560bc4d6564d6ae54
Status: Image is up to date for jasonrivers/nagios:latest
docker.io/jasonrivers/nagios:latest

C:\Users\andam>docker run --name nagiosdemo -p 8888:80 jasonrivers/nagios:latest
docker: Error response from daemon: Conflict. The container name "/nagiosdemo" is already in use by container "b95db79f28e087777c25afcdcd8a99d44ba77643612d0ad28ee693cd9ec04422". You have to remove (or rename) that container to be able to reuse that name.
See 'docker run --help'.

C:\Users\andam>docker run --name nagiosdemos -p 8888:80 jasonrivers/nagios:latest
Adding password for user nagiosadmin
chown: warning: '.' should be ':' 'nagios.nagios'
Started runsndir, PID is 13
checking permissions for nagios & nagiosgraph
rsyslogd: [origin software="rsyslogd" swVersion="8.2312.0" x-pid="21" x-info="https://www.rsyslog.com"] start

Nagios Core 4.5.7
Copyright (c) 2009-present Nagios Core Development Team and Community Contributors
Copyright (c) 1999-2009 Ethan Galstad
Last Modified: 2024-10-24
```

Step 3: Access Nagios Dashboard



Copyright © 2010-2024 Nagios Core Development Team and Community Contributors. Copyright © 1999-2009 Ethan Galstad. See the THANKS file for more information on contributors.

Nagios Core is licensed under the GNU General Public License and is provided AS IS with NO WARRANTY OF ANY KIND, INCLUDING THE WARRANTY OF DESIGN, MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Nagios, Nagios Core and the Nagios logo are trademarks, service marks, registered trademarks or service marks of Nagios Enterprises, LLC. Use of the Nagios marks is governed by the trademark use restrictions.

https://www.nagios.org/launch?utm_campaign=csp&utm_source=nagioscore&utm_medium=splash_thumbnail&utm_content=4.5.7 by Nagios Enterprises, LLC. Use of the Nagios marks is governed by the trademark use restrictions.

Step 4: Monitoring Host Details

The screenshot shows the Nagios web interface at localhost:3888. The top navigation bar includes tabs for "minikube install command - Go" and "Nagios: localhost". The main content area displays the "Current Network Status" with last updated information and version 4.5.7. It shows "Host Status Totals" with 1 Up, 0 Down, 0 Unreachable, and 0 Pending hosts. The "Service Status Totals" show 2 Ok, 0 Warning, 0 Unknown, 0 Critical, and 5 Pending services. Below these are links for service status detail, host status overview, host status summary, and host status grid.

The central part of the page is titled "Host Status Details For All Host Groups" with a limit of 100 results. It lists one host, "localhost", which is UP and was last checked on 10-22-2025 10:51:05. The status information indicates PING OK - Packet loss = 0%, RTA = 0.06 ms.

The left sidebar contains several sections:

- General**: Home, Documentation, Current Status (selected).
- Tactical Overview**: Map, Hosts, Services, Host Groups, Summary, Grid, Service Groups, Summary, Grid, Problems, Services (Unhandled), Hosts (Unhandled), Network Outages.
- Reports**: Availability, Trends, Alerts, History, Summary, Histogram, Notifications, Event Log.
- System**: Comments, Downtime, Process Info, Performance Info.

A "Quick Search" input field is also present in the sidebar.

SBQ's

1. Your Pod keeps restarting repeatedly. What will you do?

Describe the pod address the issue and solve it or delete the pod

2. A Kubernetes pod is stuck in a "Pending" state. What could be the possible reasons, and how would you troubleshoot it?

Reasons may be insufficient resources

Image pull issue

Troubleshoot:

Describe and check FailedScheduling

3. How would you debug a failed deployment in Kubernetes?

kubectl get deployment deployment-name

kubectl describe deployment deployment-name

And check logs

4. You have a Kubernetes Deployment with multiple replicas, and some pods are failing health checks. How would you identify the root cause and fix it?

Describe the pod and check liveness and readiness probes

5. How do you roll back a faulty deployment?

kubectl rollout undo deployment/deployment-name

6. How do you debug a running Pod?

kubectl logs pod-name

7. You need to expose a local service externally. How to do?

kubectl expose deployment myapp --type=NodePort --port=80 --target-port=80

8. How to Start and stop Nagios

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

docker stop nagiosdemo

docker rm nagiosdemo

9. You installed Nagios but the web interface shows "*Unable to connect to Nagios process*". how to resolve this?

Verify whether is running

ps -ef | grep nagios

10. You added a new host in Nagios, but it's not appearing on the web interface. how to check?

Verify conf

nagios -v /usr/local/nagios/etc/nagios.cfg

11. How can you check whether Nagios is running properly?

docker ps

or web access

12. How do you view Nagios logs in real-time

```
docker logs -f nagiosdemo
```

13. What are the advantages of using Nagios?

Continuous monitoring of servers, services, and applications