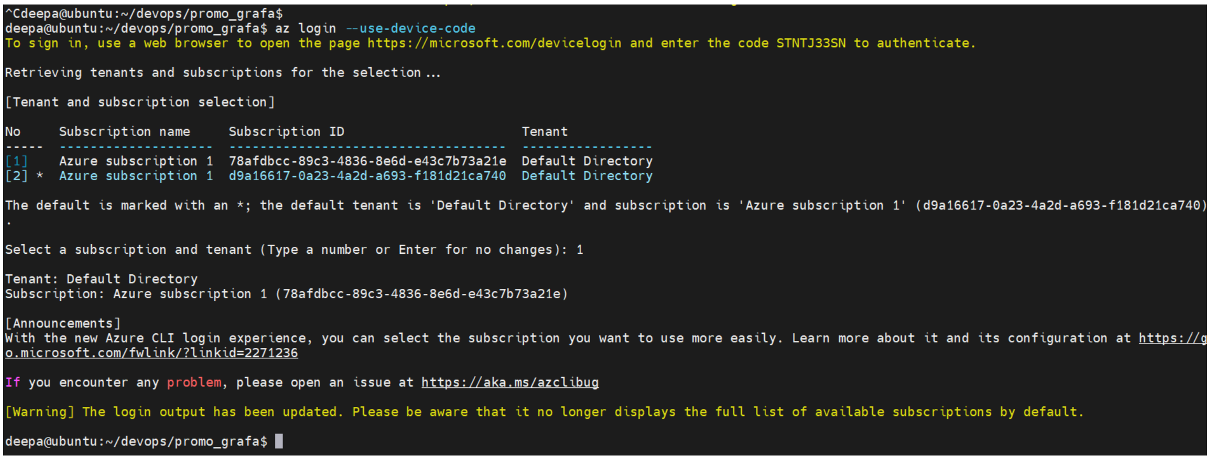
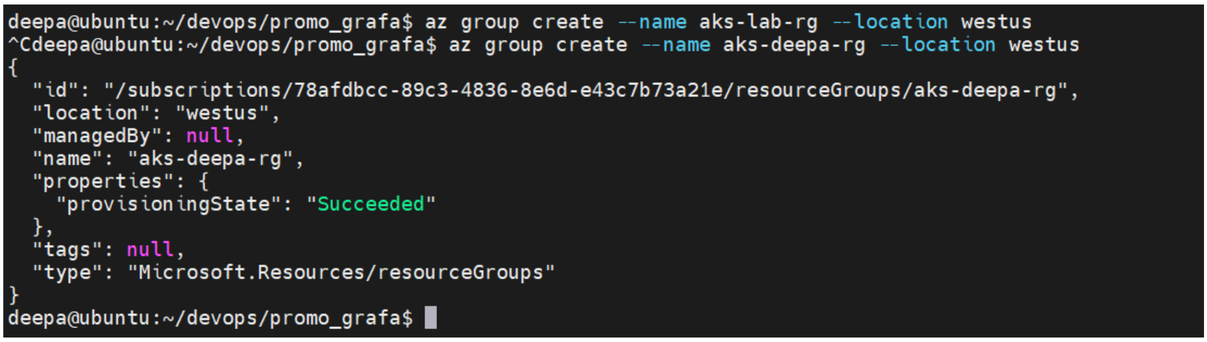
Explore on Prometheus and Grafana:

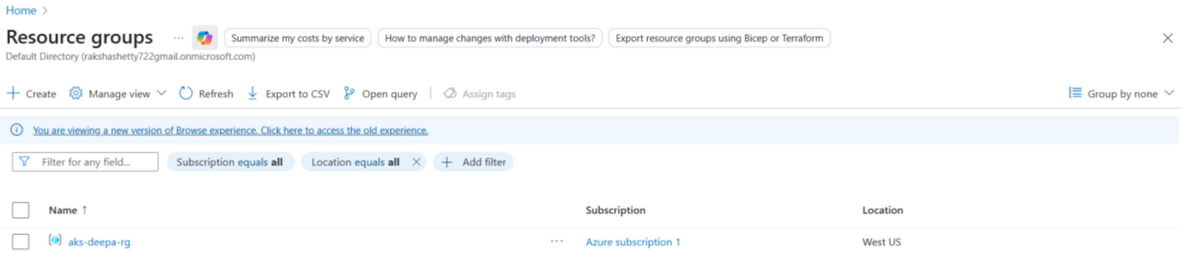


```
^Cdeepa@ubuntu:~/devops/promo_grafa$
deepa@ubuntu:~/devops/promo_grafa$ az login --use-device-code
To sign in, use a web browser to open the page https://microsoft.com/devicelogin and enter the code STNTJ33SN to authenticate.

Retrieving tenants and subscriptions for the selection ...

[Tenant and subscription selection]

No     Subscription name     Subscription ID                       Tenant
-----  --------------------  ------------------------------------  -----------------
[1]    Azure subscription 1  78afdbcc-89c3-4836-8e6d-e43c7b73a21e  Default Directory
[2] *  Azure subscription 1  d9a16617-0a23-4a2d-a693-f181d21ca740  Default Directory

The default is marked with an *; the default tenant is 'Default Directory' and subscription is 'Azure subscription 1' (d9a16617-0a23-4a2d-a693-f181d21ca740)
.

Select a subscription and tenant (Type a number or Enter for no changes): 1

Tenant: Default Directory
Subscription: Azure subscription 1 (78afdbcc-89c3-4836-8e6d-e43c7b73a21e)

[Announcements]
With the new Azure CLI login experience, you can select the subscription you want to use more easily. Learn more about it and its configuration at https://g
o.microsoft.com/fwlink/?linkid=2271236

If you encounter any problem, please open an issue at https://aka.ms/azclibug

[Warning] The login output has been updated. Please be aware that it no longer displays the full list of available subscriptions by default.

deepa@ubuntu:~/devops/promo_grafa$
```

Creating resource group in Azure cloud:



```
deepa@ubuntu:~/devops/promo_grafa$ az group create --name aks-lab-rg --location westus
^Cdeepa@ubuntu:~/devops/promo_grafa$ az group create --name aks-deepa-rg --location westus
{
  "id": "/subscriptions/78afdbcc-89c3-4836-8e6d-e43c7b73a21e/resourceGroups/aks-deepa-rg",
  "location": "westus",
  "managedBy": null,
  "name": "aks-deepa-rg",
  "properties": {
    "provisioningState": "Succeeded"
  },
  "tags": null,
  "type": "Microsoft.Resources/resourceGroups"
}
deepa@ubuntu:~/devops/promo_grafa$
```

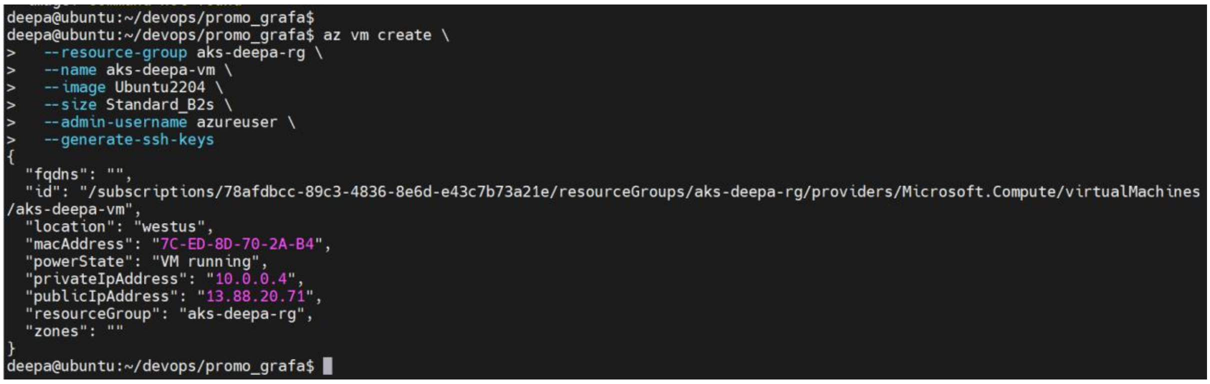


Default Directory (rakshashetty722gmail.onmicrosoft.com)

You are viewing a new version of Browse experience. Click here to access the old experience.

□	Name ↑	Subscription	Location
□	aks-deepa-rg	Azure subscription 1	West US

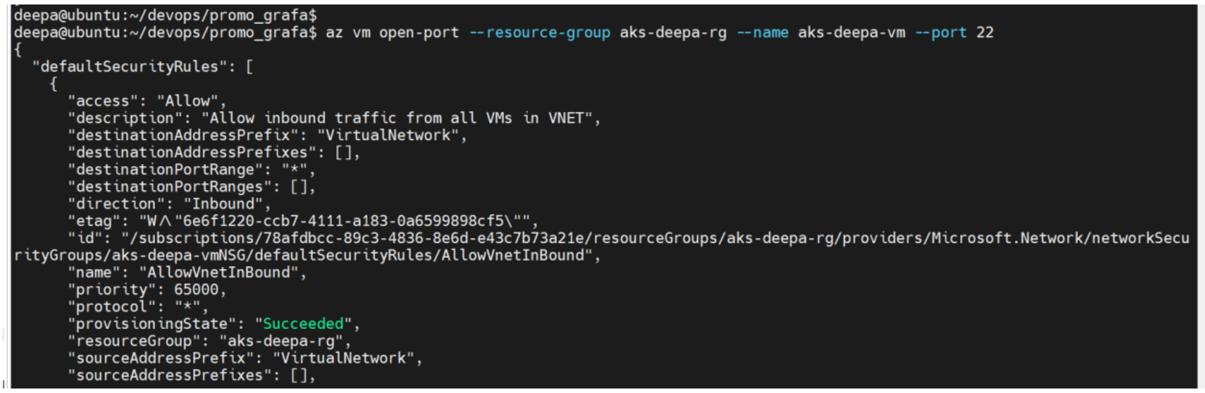
Create the azure VM:



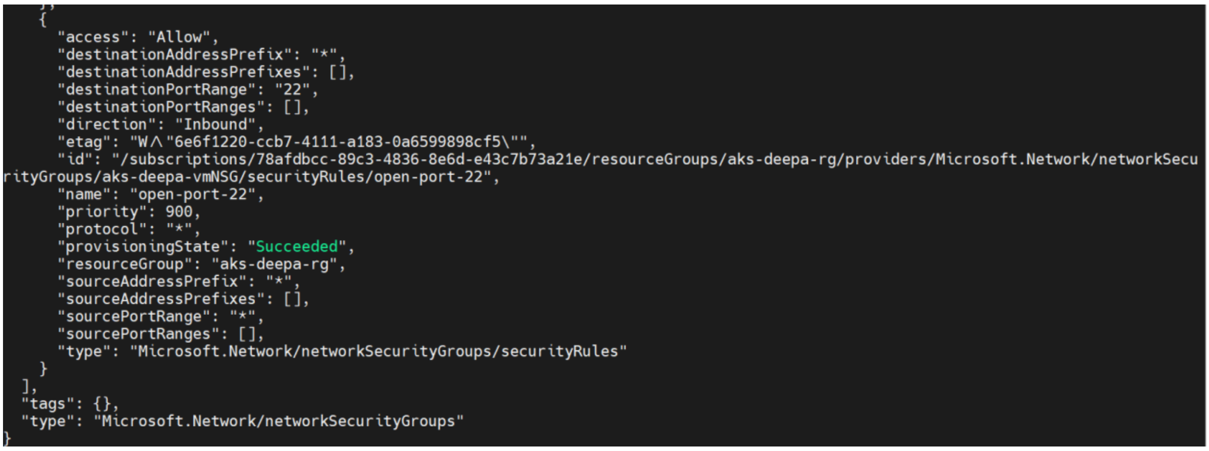
```
deepa@ubuntu:~/devops/promo_grafa$
deepa@ubuntu:~/devops/promo_grafa$ az vm create \
>   --resource-group aks-deepa-rg \
>   --name aks-deepa-vm \
>   --image Ubuntu2204 \
>   --size Standard_B2s \
>   --admin-username azureuser \
>   --generate-ssh-keys
{
  "fqdns": "",
  "id": "/subscriptions/78afdbcc-89c3-4836-8e6d-e43c7b73a21e/resourceGroups/aks-deepa-rg/providers/Microsoft.Compute/virtualMachines
/aks-deepa-vm",
  "location": "westus",
  "macAddress": "7C-ED-8D-70-2A-B4",
  "powerState": "VM running",
  "privateIpAddress": "10.0.0.4",
  "publicIpAddress": "13.88.20.71",
  "resourceGroup": "aks-deepa-rg",
  "zones": ""
}
deepa@ubuntu:~/devops/promo_grafa$
```

Open SSH and web ports (Grafana/Prometheus):

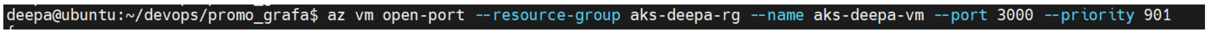
Specify a Unique Priority for Each New Rule



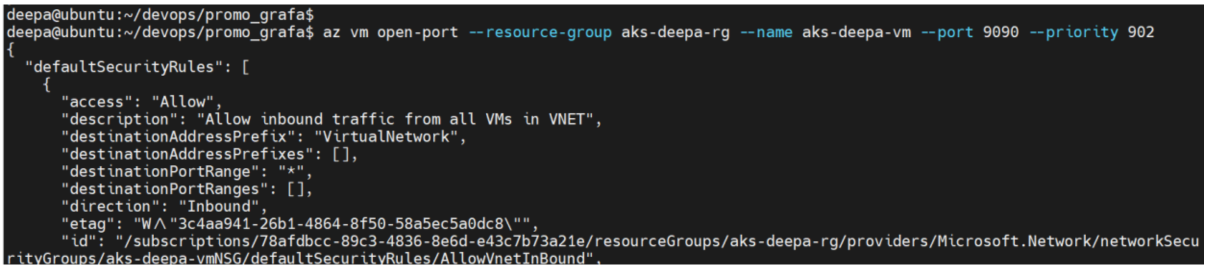
```
deepa@ubuntu:~/devops/promo_grafa$
deepa@ubuntu:~/devops/promo_grafa$ az vm open-port --resource-group aks-deepa-rg --name aks-deepa-vm --port 22
{
  "defaultSecurityRules": [
    {
      "access": "Allow",
      "description": "Allow inbound traffic from all VMs in VNET",
      "destinationAddressPrefix": "VirtualNetwork",
      "destinationAddressPrefixes": [],
      "destinationPortRange": "*",
      "destinationPortRanges": [],
      "direction": "Inbound",
      "etag": "W/\"6e6f1220-ccb7-4111-a183-0a6599898cf5\"",
      "id": "/subscriptions/78afdbcc-89c3-4836-8e6d-e43c7b73a21e/resourceGroups/aks-deepa-rg/providers/Microsoft.Network/networkSecu
rityGroups/aks-deepa-vmNSG/defaultSecurityRules/AllowVnetInBound",
      "name": "AllowVnetInBound",
      "priority": 65000,
      "protocol": "*",
      "provisioningState": "Succeeded",
      "resourceGroup": "aks-deepa-rg",
      "sourceAddressPrefix": "VirtualNetwork",
      "sourceAddressPrefixes": [],
```



```
    },
    {
      "access": "Allow",
      "destinationAddressPrefix": "*",
      "destinationAddressPrefixes": [],
      "destinationPortRange": "22",
      "destinationPortRanges": [],
      "direction": "Inbound",
      "etag": "W/\"6e6f1220-ccb7-4111-a183-0a6599898cf5\"",
      "id": "/subscriptions/78afdbcc-89c3-4836-8e6d-e43c7b73a21e/resourceGroups/aks-deepa-rg/providers/Microsoft.Network/networkSecu
rityGroups/aks-deepa-vmNSG/securityRules/open-port-22",
      "name": "open-port-22",
      "priority": 900,
      "protocol": "*",
      "provisioningState": "Succeeded",
      "resourceGroup": "aks-deepa-rg",
      "sourceAddressPrefix": "*",
      "sourceAddressPrefixes": [],
      "sourcePortRange": "*",
      "sourcePortRanges": [],
      "type": "Microsoft.Network/networkSecurityGroups/securityRules"
    }
  ],
  "tags": {},
  "type": "Microsoft.Network/networkSecurityGroups"
}
```

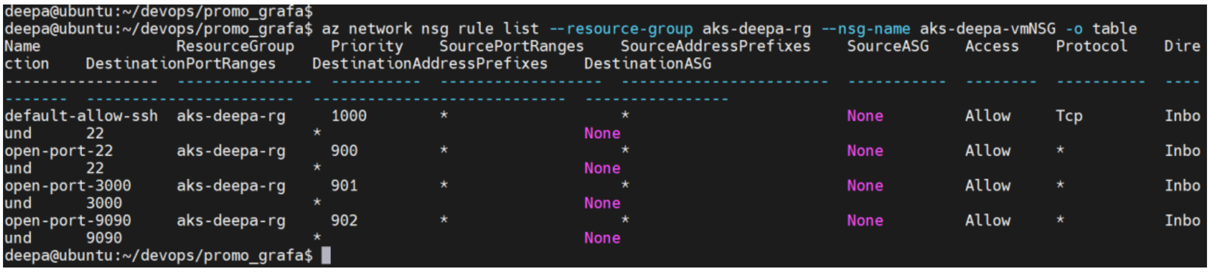


```
deepa@ubuntu:~/devops/promo_grafa$ az vm open-port --resource-group aks-deepa-rg --name aks-deepa-vm --port 3000 --priority 901
```



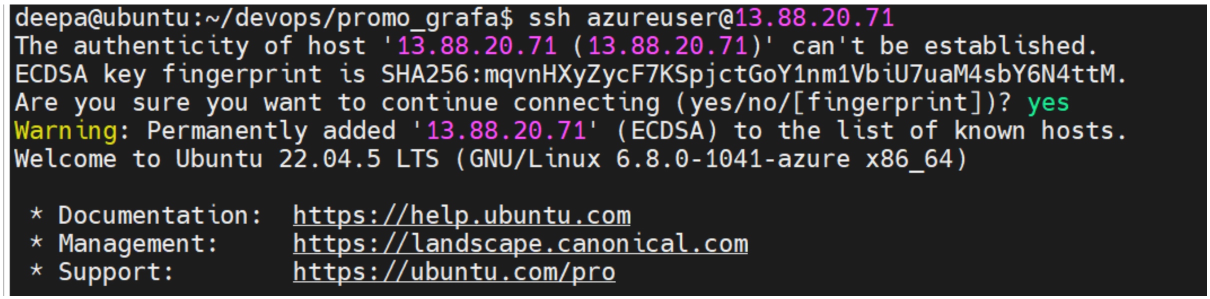
```
deepa@ubuntu:~/devops/promo_grafa$
deepa@ubuntu:~/devops/promo_grafa$ az vm open-port --resource-group aks-deepa-rg --name aks-deepa-vm --port 9090 --priority 902
{
  "defaultSecurityRules": [
    {
      "access": "Allow",
      "description": "Allow inbound traffic from all VMs in VNET",
      "destinationAddressPrefix": "VirtualNetwork",
      "destinationAddressPrefixes": [],
      "destinationPortRange": "*",
      "destinationPortRanges": [],
      "direction": "Inbound",
      "etag": "W/\"3c4aa941-26b1-4864-8f50-58a5ec5a0dc8\"",
      "id": "/subscriptions/78afdbcc-89c3-4836-8e6d-e43c7b73a21e/resourceGroups/aks-deepa-rg/providers/Microsoft.Network/networkSecu
rityGroups/aks-deepa-vmNSG/defaultSecurityRules/AllowVnetInBound",
```

Verify Rules:

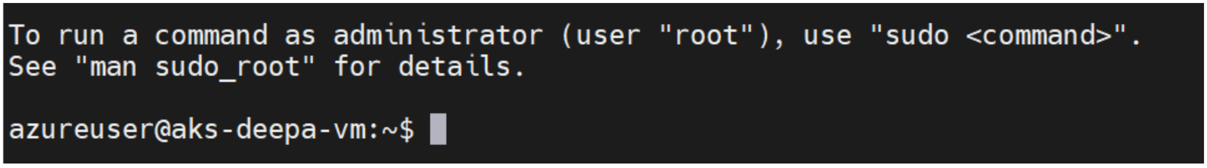


```
deepa@ubuntu:~/devops/promo_grafa$
deepa@ubuntu:~/devops/promo_grafa$ az network nsg rule list --resource-group aks-deepa-rg --nsg-name aks-deepa-vmNSG -o table
Name                ResourceGroup    Priority    SourcePortRanges    SourceAddressPrefixes    SourceASG    Access    Protocol    Dire
ction    DestinationPortRanges    DestinationAddressPrefixes    DestinationASG
-----------------  ---------------  ----------  ------------------  -----------------------  -----------  --------  ----------  ----
-------  -----------------------  ----------------------------  ----------------
default-allow-ssh  aks-deepa-rg     1000        *                   *                        None         Allow     Tcp         Inbo
und      22                       *                             None
open-port-22       aks-deepa-rg     900         *                   *                        None         Allow     *           Inbo
und      22                       *                             None
open-port-3000     aks-deepa-rg     901         *                   *                        None         Allow     *           Inbo
und      3000                     *                             None
open-port-9090     aks-deepa-rg     902         *                   *                        None         Allow     *           Inbo
und      9090                     *                             None
deepa@ubuntu:~/devops/promo_grafa$
```

SSH into your VM:

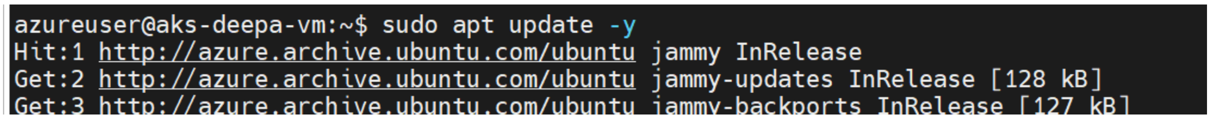


```
deepa@ubuntu:~/devops/promo_grafa$ ssh azureuser@13.88.20.71
The authenticity of host '13.88.20.71 (13.88.20.71)' can't be established.
ECDSA key fingerprint is SHA256:mqvnHXyZycF7KSpjctGoY1nm1VbiU7uaM4sbY6N4ttM.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '13.88.20.71' (ECDSA) to the list of known hosts.
Welcome to Ubuntu 22.04.5 LTS (GNU/Linux 6.8.0-1041-azure x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro
```

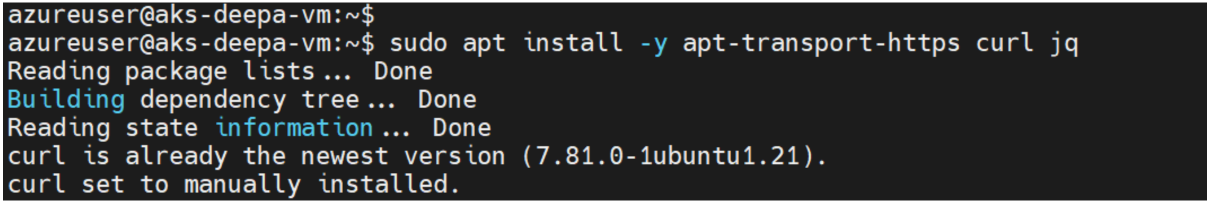


```
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

azureuser@aks-deepa-vm:~$
```

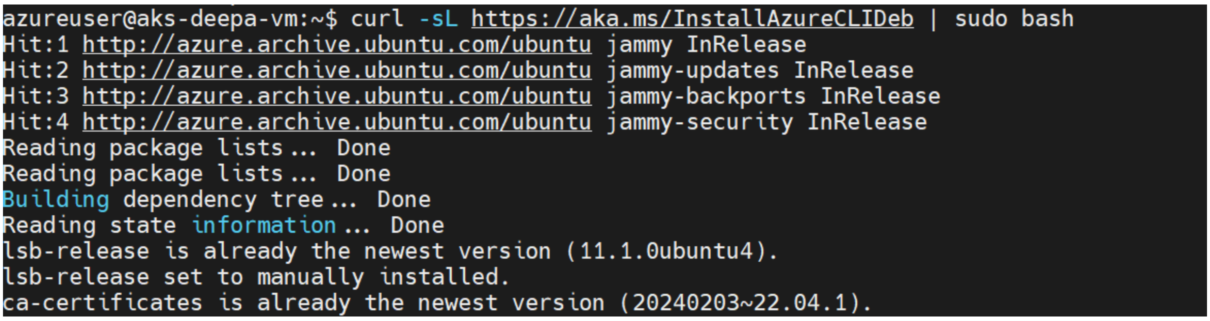
Install Required Tools on the VM:



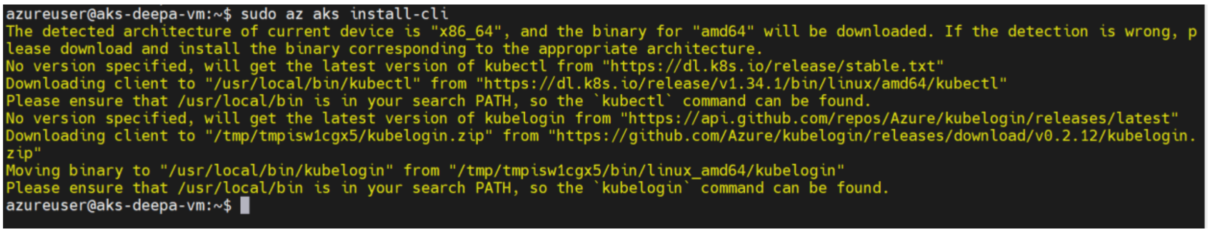
```
azureuser@aks-deepa-vm:~$ sudo apt update -y
Hit:1 http://azure.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://azure.archive.ubuntu.com/ubuntu jammy-updates InRelease [128 kB]
Get:3 http://azure.archive.ubuntu.com/ubuntu jammy-backports InRelease [127 kB]
```



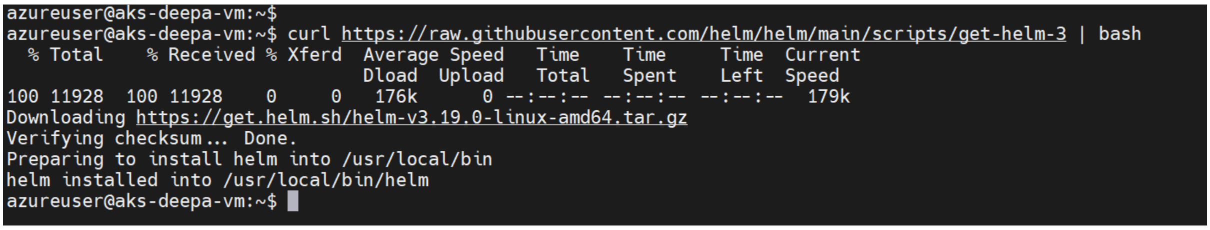
```
azureuser@aks-deepa-vm:~$
azureuser@aks-deepa-vm:~$ sudo apt install -y apt-transport-https curl jq
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
curl is already the newest version (7.81.0-1ubuntu1.21).
curl set to manually installed.
```



```
azureuser@aks-deepa-vm:~$ curl -sL https://aka.ms/InstallAzureCLIDeb | sudo bash
Hit:1 http://azure.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://azure.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://azure.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:4 http://azure.archive.ubuntu.com/ubuntu jammy-security InRelease
Reading package lists... Done
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
lsb-release is already the newest version (11.1.0ubuntu4).
lsb-release set to manually installed.
ca-certificates is already the newest version (20240203~22.04.1).
```

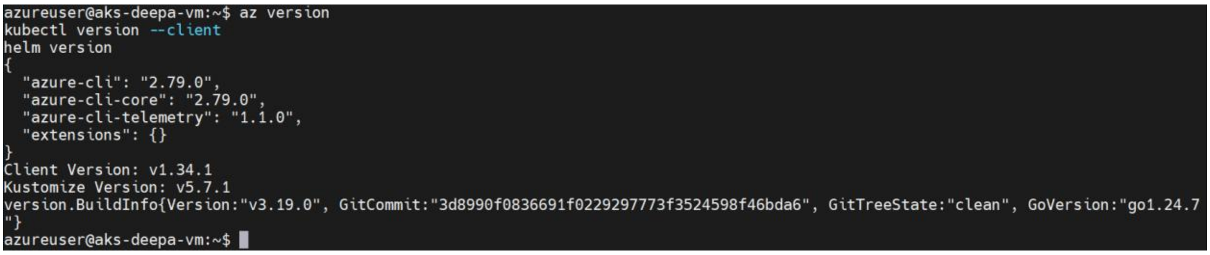


```
azureuser@aks-deepa-vm:~$ sudo az aks install-cli
The detected architecture of current device is "x86_64", and the binary for "amd64" will be downloaded. If the detection is wrong, p
lease download and install the binary corresponding to the appropriate architecture.
No version specified, will get the latest version of kubectl from "https://dl.k8s.io/release/stable.txt"
Downloading client to "/usr/local/bin/kubectl" from "https://dl.k8s.io/release/v1.34.1/bin/linux/amd64/kubectl"
Please ensure that /usr/local/bin is in your search PATH, so the `kubectl` command can be found.
No version specified, will get the latest version of kubelogin from "https://api.github.com/repos/Azure/kubelogin/releases/latest"
Downloading client to "/tmp/tmpisw1cgx5/kubelogin.zip" from "https://github.com/Azure/kubelogin/releases/download/v0.2.12/kubelogin.
zip"
Moving binary to "/usr/local/bin/kubelogin" from "/tmp/tmpisw1cgx5/bin/linux_amd64/kubelogin"
Please ensure that /usr/local/bin is in your search PATH, so the `kubelogin` command can be found.
azureuser@aks-deepa-vm:~$
```



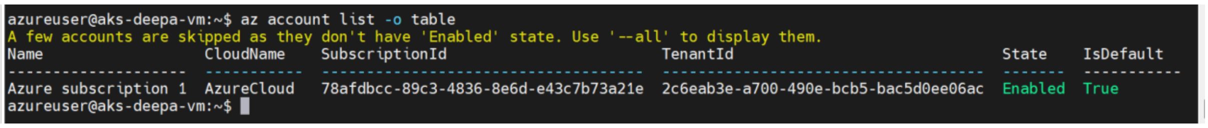
```
azureuser@aks-deepa-vm:~$
azureuser@aks-deepa-vm:~$ curl https://raw.githubusercontent.com/helm/helm/main/scripts/get-helm-3 | bash
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 11928  100 11928    0     0   176k      0 --:--:-- --:--:-- --:--:--  179k
Downloading https://get.helm.sh/helm-v3.19.0-linux-amd64.tar.gz
Verifying checksum... Done.
Preparing to install helm into /usr/local/bin
helm installed into /usr/local/bin/helm
azureuser@aks-deepa-vm:~$
```

Verify:



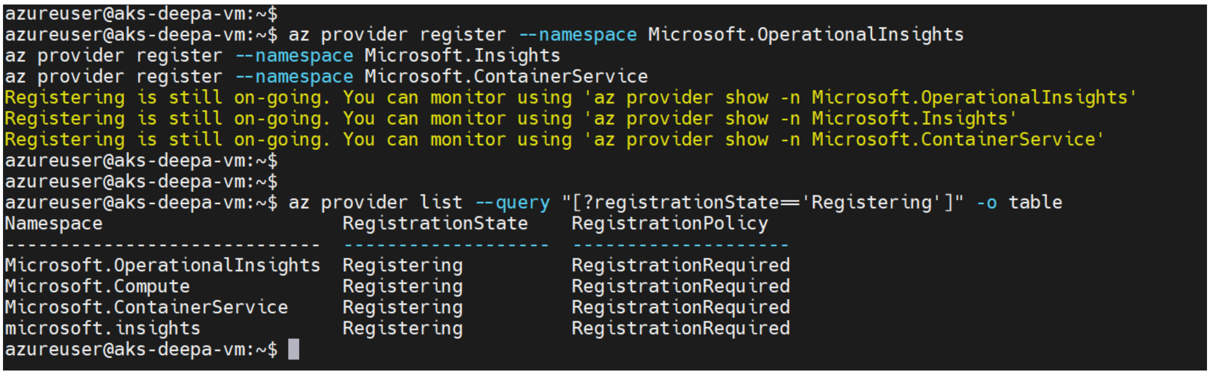
```
azureuser@aks-deepa-vm:~$ az version
kubectl version --client
helm version
{
  "azure-cli": "2.79.0",
  "azure-cli-core": "2.79.0",
  "azure-cli-telemetry": "1.1.0",
  "extensions": {}
}
Client Version: v1.34.1
Kustomize Version: v5.7.1
version.BuildInfo{Version:"v3.19.0", GitCommit:"3d8990f0836691f0229297773f3524598f46bda6", GitTreeState:"clean", GoVersion:"go1.24.7
"}
azureuser@aks-deepa-vm:~$
```

Create AKS cluster:



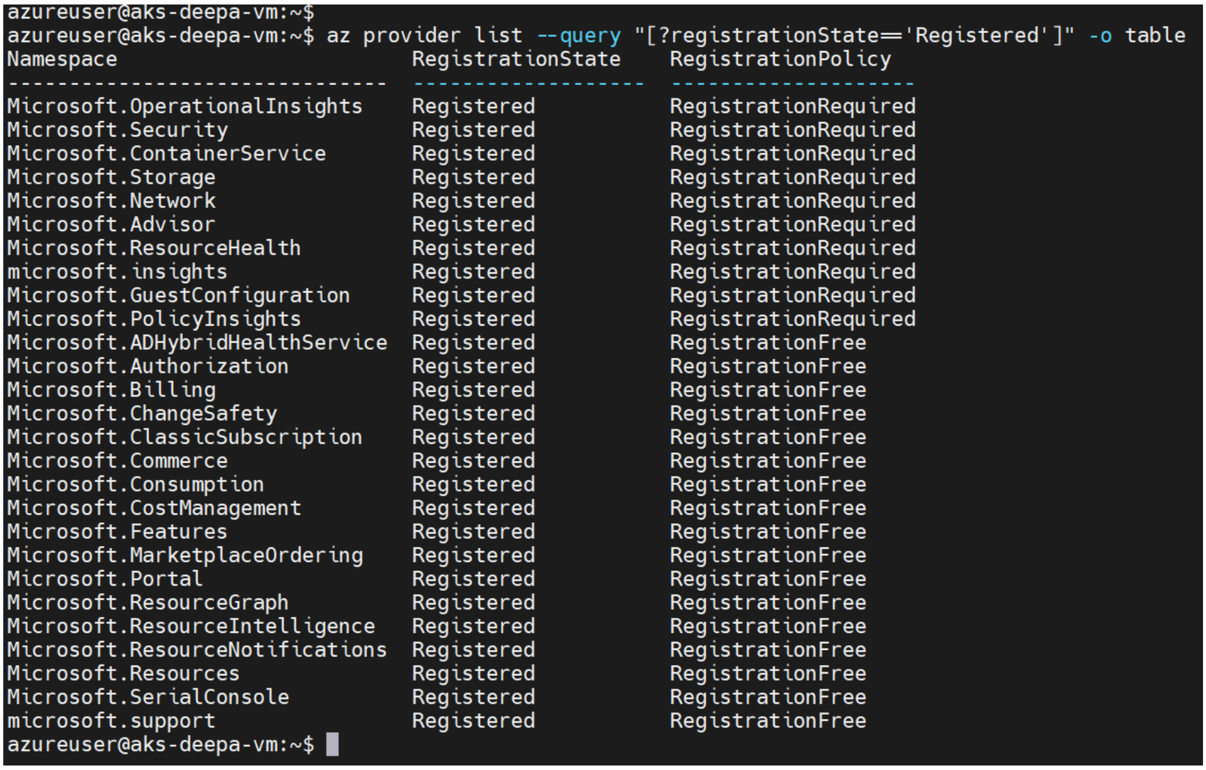
```
azureuser@aks-deepa-vm:~$ az account list -o table
A few accounts are skipped as they don't have 'Enabled' state. Use '--all' to display them.
Name                  CloudName    SubscriptionId                        TenantId                              State    IsDefault
--------------------  -----------  ------------------------------------  ------------------------------------  -------  -----------
Azure subscription 1  AzureCloud   78afdbcc-89c3-4836-8e6d-e43c7b73a21e  2c6eab3e-a700-490e-bcb5-bac5d0ee06ac  Enabled  True
azureuser@aks-deepa-vm:~$
```

Register the Missing Providers:



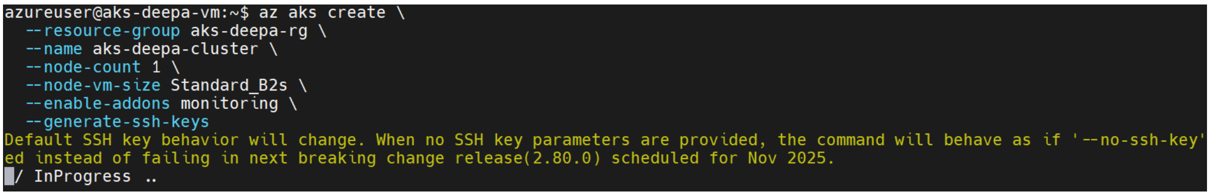
```
azureuser@aks-deepa-vm:~$
azureuser@aks-deepa-vm:~$ az provider register --namespace Microsoft.OperationalInsights
az provider register --namespace Microsoft.Insights
az provider register --namespace Microsoft.ContainerService
Registering is still on-going. You can monitor using 'az provider show -n Microsoft.OperationalInsights'
Registering is still on-going. You can monitor using 'az provider show -n Microsoft.Insights'
Registering is still on-going. You can monitor using 'az provider show -n Microsoft.ContainerService'
azureuser@aks-deepa-vm:~$
azureuser@aks-deepa-vm:~$
azureuser@aks-deepa-vm:~$ az provider list --query "[?registrationState=='Registering']" -o table
Namespace                      RegistrationState    RegistrationPolicy
-----------------------------  -------------------  --------------------
Microsoft.OperationalInsights  Registering          RegistrationRequired
Microsoft.Compute              Registering          RegistrationRequired
Microsoft.ContainerService     Registering          RegistrationRequired
microsoft.insights             Registering          RegistrationRequired
azureuser@aks-deepa-vm:~$
```

Verify they're all "Registered":

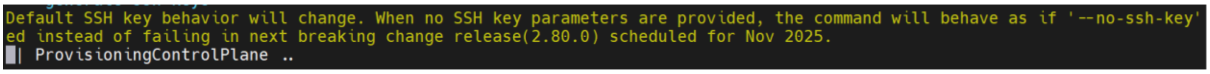


```
azureuser@aks-deepa-vm:~$
azureuser@aks-deepa-vm:~$ az provider list --query "[?registrationState=='Registered']" -o table
Namespace                          RegistrationState    RegistrationPolicy
---------------------------------  -------------------  --------------------
Microsoft.OperationalInsights      Registered           RegistrationRequired
Microsoft.Security                 Registered           RegistrationRequired
Microsoft.ContainerService         Registered           RegistrationRequired
Microsoft.Storage                  Registered           RegistrationRequired
Microsoft.Network                  Registered           RegistrationRequired
Microsoft.Advisor                  Registered           RegistrationRequired
Microsoft.ResourceHealth           Registered           RegistrationRequired
microsoft.insights                 Registered           RegistrationRequired
Microsoft.GuestConfiguration       Registered           RegistrationRequired
Microsoft.PolicyInsights           Registered           RegistrationRequired
Microsoft.ADHybridHealthService    Registered           RegistrationFree
Microsoft.Authorization            Registered           RegistrationFree
Microsoft.Billing                  Registered           RegistrationFree
Microsoft.ChangeSafety             Registered           RegistrationFree
Microsoft.ClassicSubscription      Registered           RegistrationFree
Microsoft.Commerce                 Registered           RegistrationFree
Microsoft.Consumption              Registered           RegistrationFree
Microsoft.CostManagement           Registered           RegistrationFree
Microsoft.Features                 Registered           RegistrationFree
Microsoft.MarketplaceOrdering      Registered           RegistrationFree
Microsoft.Portal                   Registered           RegistrationFree
Microsoft.ResourceGraph            Registered           RegistrationFree
Microsoft.ResourceIntelligence     Registered           RegistrationFree
Microsoft.ResourceNotifications    Registered           RegistrationFree
Microsoft.Resources                Registered           RegistrationFree
Microsoft.SerialConsole            Registered           RegistrationFree
microsoft.support                  Registered           RegistrationFree
azureuser@aks-deepa-vm:~$
```

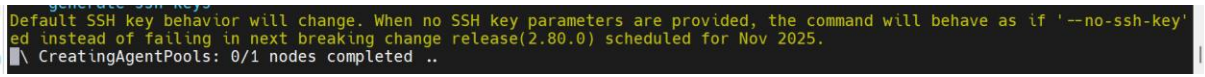
Retry AKS Creation:



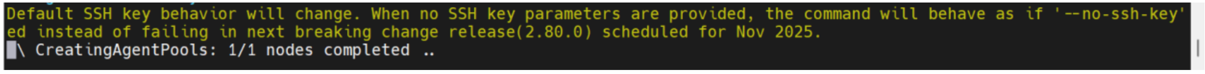
```
azureuser@aks-deepa-vm:~$ az aks create \
  --resource-group aks-deepa-rg \
  --name aks-deepa-cluster \
  --node-count 1 \
  --node-vm-size Standard_B2s \
  --enable-addons monitoring \
  --generate-ssh-keys
Default SSH key behavior will change. When no SSH key parameters are provided, the command will behave as if '--no-ssh-key'
ed instead of failing in next breaking change release(2.80.0) scheduled for Nov 2025.
█/ InProgress ..
```



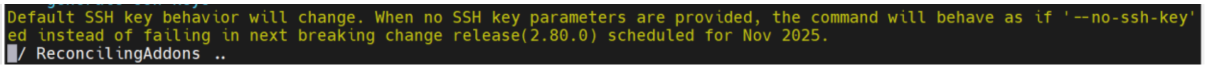
```
Default SSH key behavior will change. When no SSH key parameters are provided, the command will behave as if '--no-ssh-key'
ed instead of failing in next breaking change release(2.80.0) scheduled for Nov 2025.
█| ProvisioningControlPlane ..
```



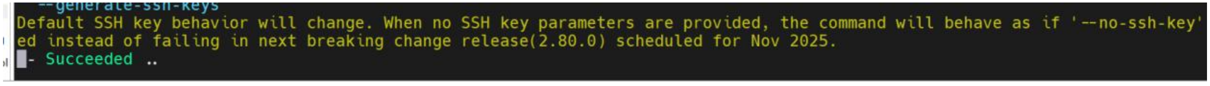
```
Default SSH key behavior will change. When no SSH key parameters are provided, the command will behave as if '--no-ssh-key'
ed instead of failing in next breaking change release(2.80.0) scheduled for Nov 2025.
█\ CreatingAgentPools: 0/1 nodes completed ..
```



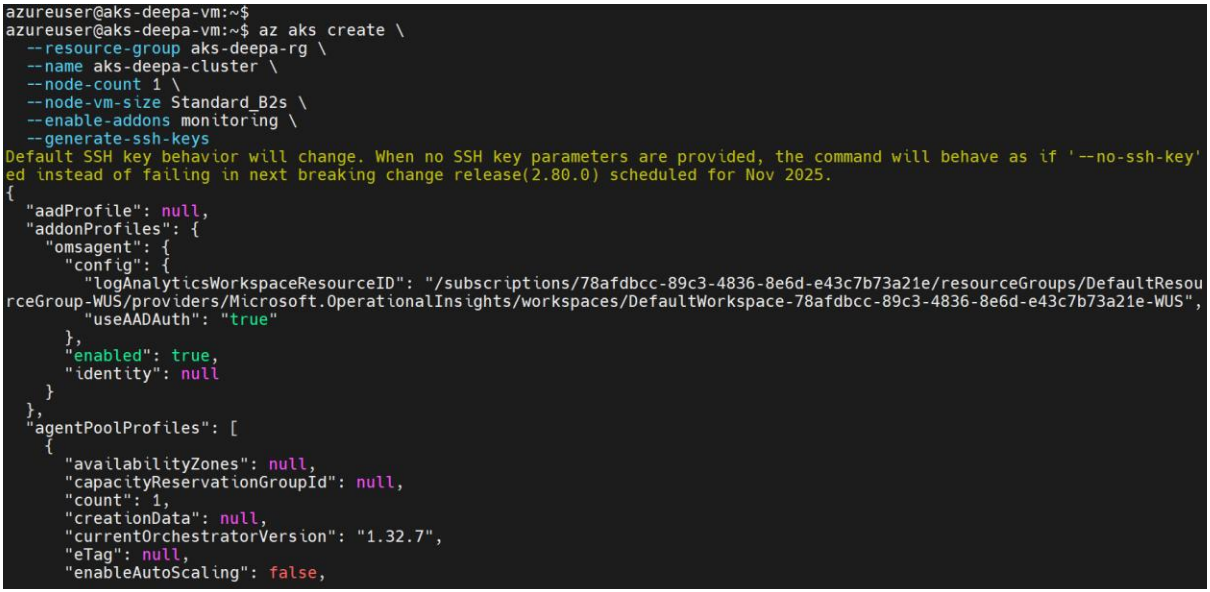
```
Default SSH key behavior will change. When no SSH key parameters are provided, the command will behave as if '--no-ssh-key'
ed instead of failing in next breaking change release(2.80.0) scheduled for Nov 2025.
█\ CreatingAgentPools: 1/1 nodes completed ..
```



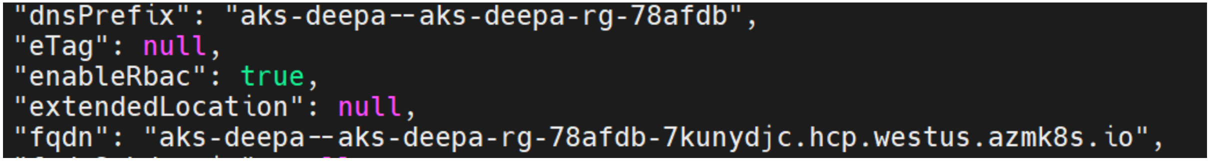
```
Default SSH key behavior will change. When no SSH key parameters are provided, the command will behave as if '--no-ssh-key'
ed instead of failing in next breaking change release(2.80.0) scheduled for Nov 2025.
█/ ReconcilingAddons ..
```



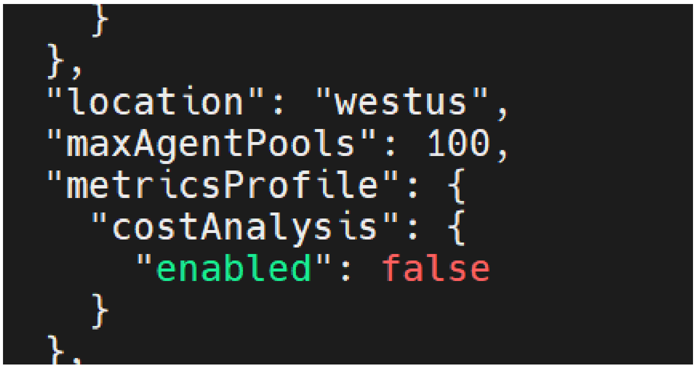
```
  --generate-ssh-keys
Default SSH key behavior will change. When no SSH key parameters are provided, the command will behave as if '--no-ssh-key'
ed instead of failing in next breaking change release(2.80.0) scheduled for Nov 2025.
█- Succeeded ..
```



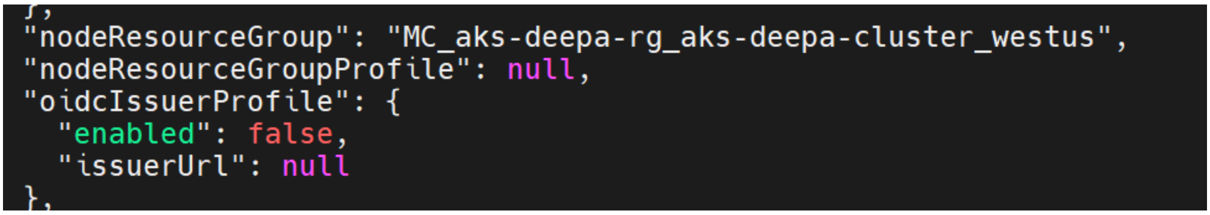
```
azureuser@aks-deepa-vm:~$
azureuser@aks-deepa-vm:~$ az aks create \
  --resource-group aks-deepa-rg \
  --name aks-deepa-cluster \
  --node-count 1 \
  --node-vm-size Standard_B2s \
  --enable-addons monitoring \
  --generate-ssh-keys
Default SSH key behavior will change. When no SSH key parameters are provided, the command will behave as if '--no-ssh-key'
ed instead of failing in next breaking change release(2.80.0) scheduled for Nov 2025.
{
  "aadProfile": null,
  "addonProfiles": {
    "omsagent": {
      "config": {
        "logAnalyticsWorkspaceResourceID": "/subscriptions/78afdbcc-89c3-4836-8e6d-e43c7b73a21e/resourceGroups/DefaultResou
rceGroup-WUS/providers/Microsoft.OperationalInsights/workspaces/DefaultWorkspace-78afdbcc-89c3-4836-8e6d-e43c7b73a21e-WUS",
        "useAADAuth": "true"
      },
      "enabled": true,
      "identity": null
    }
  },
  "agentPoolProfiles": [
    {
      "availabilityZones": null,
      "capacityReservationGroupId": null,
      "count": 1,
      "creationData": null,
      "currentOrchestratorVersion": "1.32.7",
      "eTag": null,
      "enableAutoScaling": false,
```



```
"dnsPrefix": "aks-deepa--aks-deepa-rg-78afdb",
"eTag": null,
"enableRbac": true,
"extendedLocation": null,
"fqdn": "aks-deepa--aks-deepa-rg-78afdb-7kunydjc.hcp.westus.azmk8s.io",
```

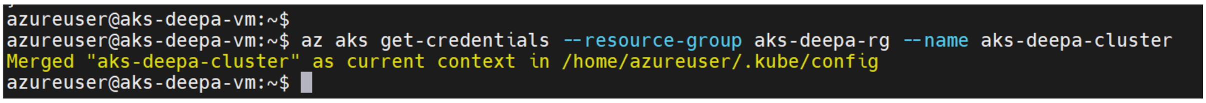


```
    }
  },
  "location": "westus",
  "maxAgentPools": 100,
  "metricsProfile": {
    "costAnalysis": {
      "enabled": false
    }
  },
```

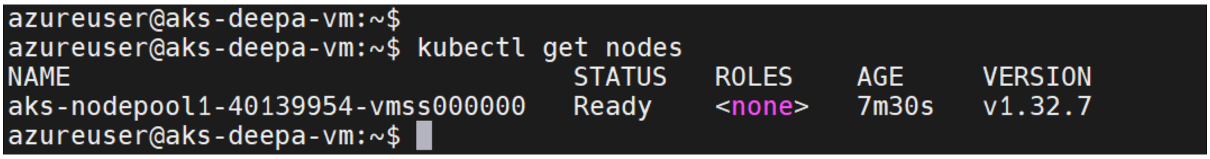


```
},
"nodeResourceGroup": "MC_aks-deepa-rg_aks-deepa-cluster_westus",
"nodeResourceGroupProfile": null,
"oidcIssuerProfile": {
  "enabled": false,
  "issuerUrl": null
},
```

Connect to Your AKS Cluster:

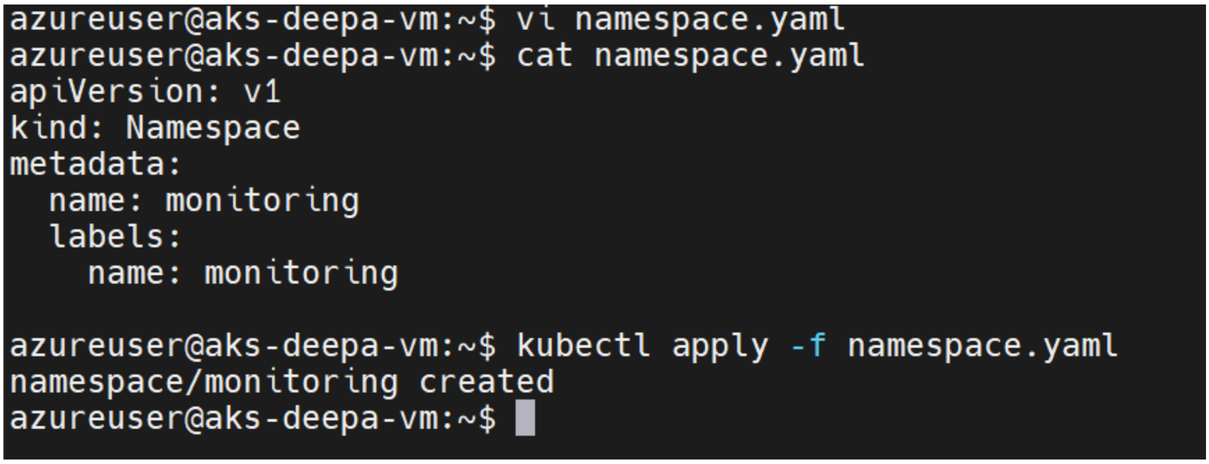


```
azureuser@aks-deepa-vm:~$
azureuser@aks-deepa-vm:~$ az aks get-credentials --resource-group aks-deepa-rg --name aks-deepa-cluster
Merged "aks-deepa-cluster" as current context in /home/azureuser/.kube/config
azureuser@aks-deepa-vm:~$
```

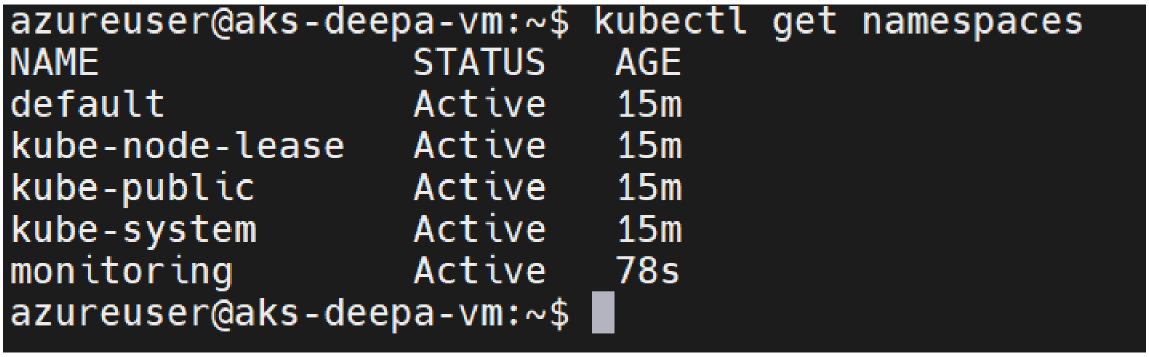


```
azureuser@aks-deepa-vm:~$
azureuser@aks-deepa-vm:~$ kubectl get nodes
NAME                                STATUS   ROLES    AGE     VERSION
aks-nodepool1-40139954-vmss000000   Ready    <none>   7m30s   v1.32.7
azureuser@aks-deepa-vm:~$
```

Create Namespace:



```
azureuser@aks-deepa-vm:~$ vi namespace.yaml
azureuser@aks-deepa-vm:~$ cat namespace.yaml
apiVersion: v1
kind: Namespace
metadata:
  name: monitoring
  labels:
    name: monitoring

azureuser@aks-deepa-vm:~$ kubectl apply -f namespace.yaml
namespace/monitoring created
azureuser@aks-deepa-vm:~$
```



```
azureuser@aks-deepa-vm:~$ kubectl get namespaces
NAME              STATUS   AGE
default           Active   15m
kube-node-lease   Active   15m
kube-public       Active   15m
kube-system       Active   15m
monitoring        Active   78s
azureuser@aks-deepa-vm:~$
```

Combined Prometheus YAML manifest that includes:

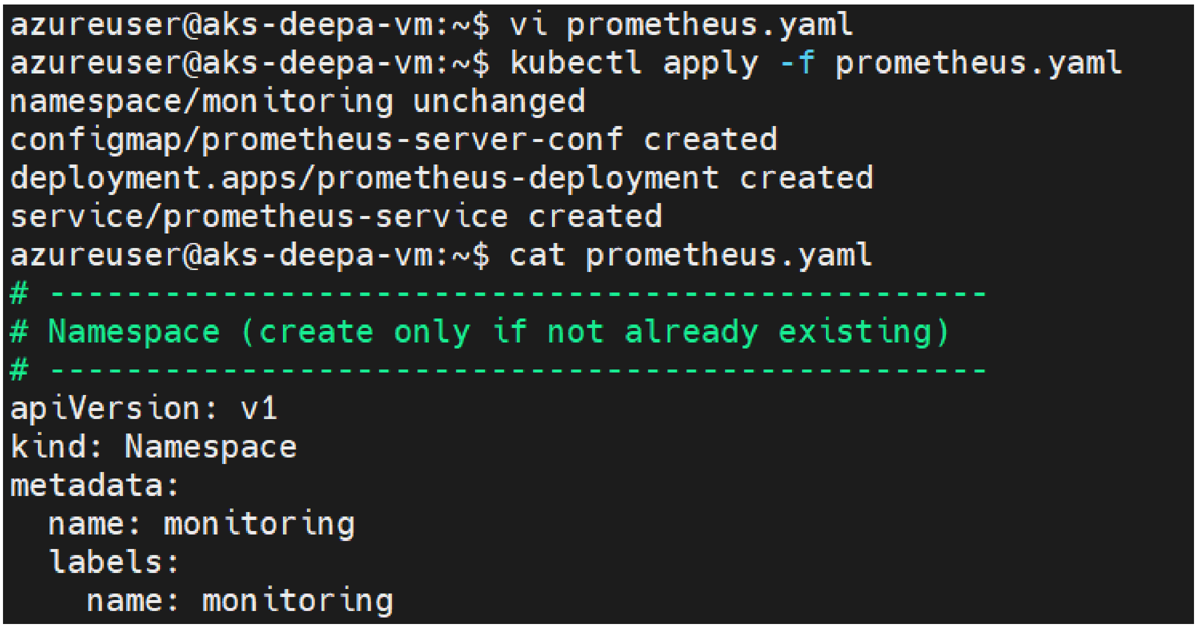
Namespace (optional, if not already created)

ConfigMap

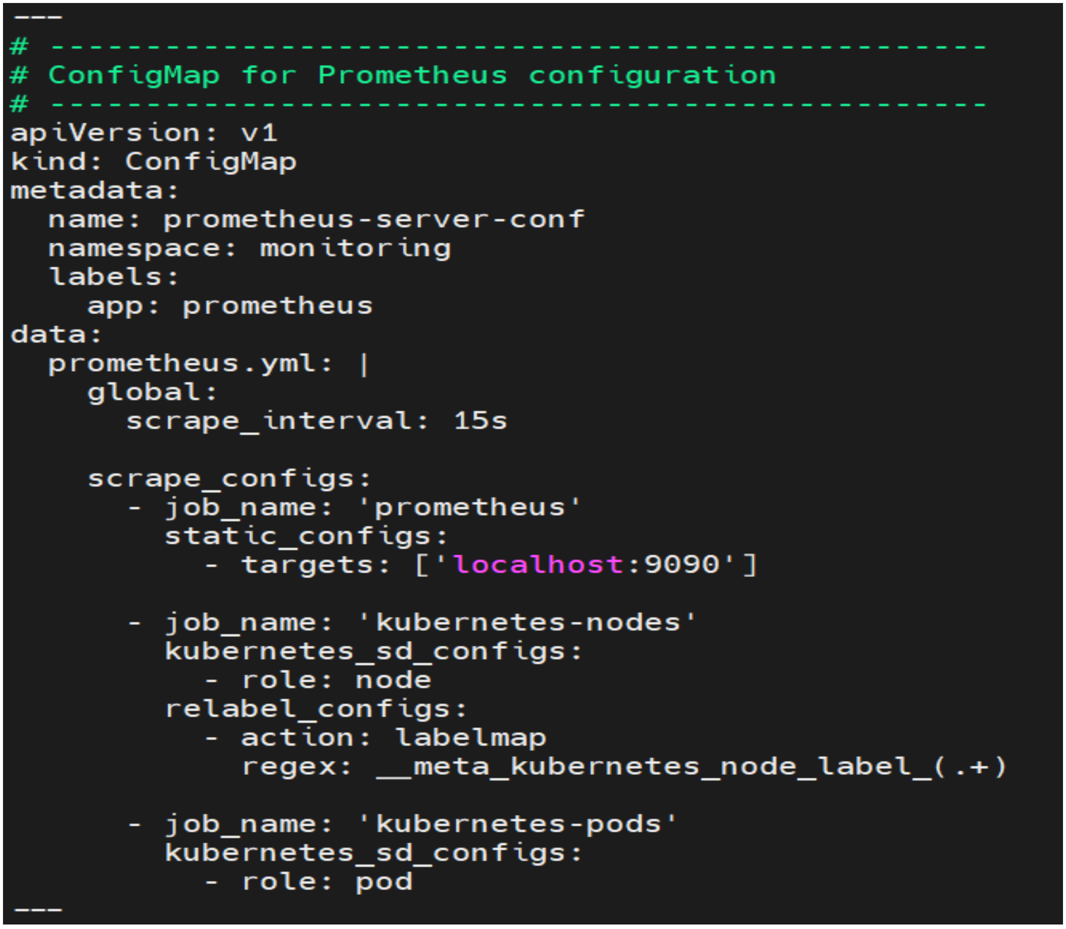
Deployment

Service

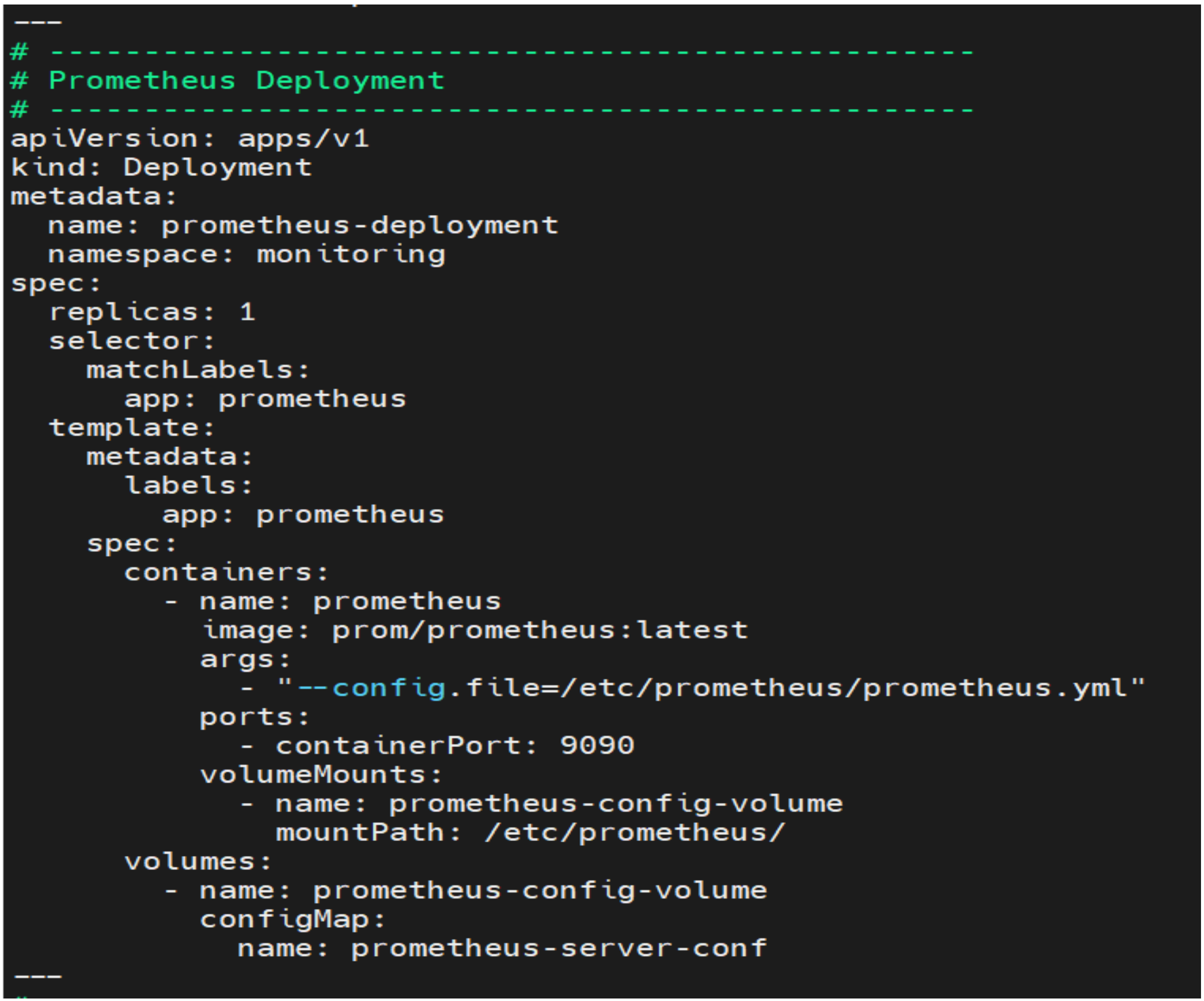
prometheus.yaml



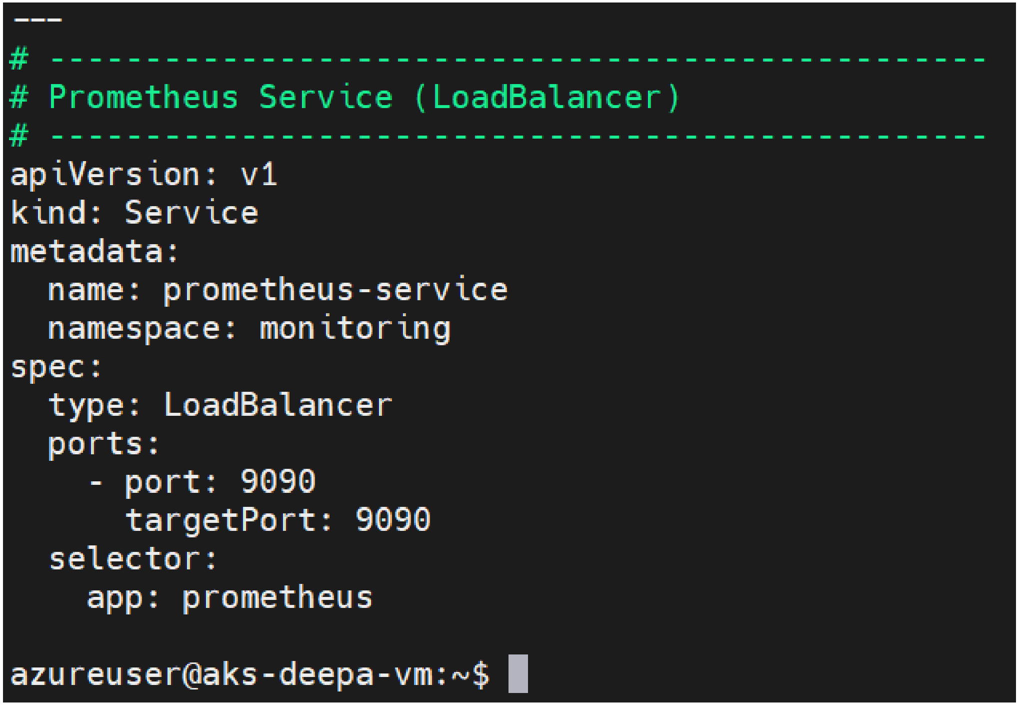
```
azureuser@aks-deepa-vm:~$ vi prometheus.yaml
azureuser@aks-deepa-vm:~$ kubectl apply -f prometheus.yaml
namespace/monitoring unchanged
configmap/prometheus-server-conf created
deployment.apps/prometheus-deployment created
service/prometheus-service created
azureuser@aks-deepa-vm:~$ cat prometheus.yaml
# -------------------------------------------------
# Namespace (create only if not already existing)
# -------------------------------------------------
apiVersion: v1
kind: Namespace
metadata:
  name: monitoring
  labels:
    name: monitoring
```



```
---
# -------------------------------------------------
# ConfigMap for Prometheus configuration
# -------------------------------------------------
apiVersion: v1
kind: ConfigMap
metadata:
  name: prometheus-server-conf
  namespace: monitoring
  labels:
    app: prometheus
data:
  prometheus.yml: |
    global:
      scrape_interval: 15s

    scrape_configs:
      - job_name: 'prometheus'
        static_configs:
          - targets: ['localhost:9090']

      - job_name: 'kubernetes-nodes'
        kubernetes_sd_configs:
          - role: node
        relabel_configs:
          - action: labelmap
            regex: __meta_kubernetes_node_label_(.+)

      - job_name: 'kubernetes-pods'
        kubernetes_sd_configs:
          - role: pod
---
```

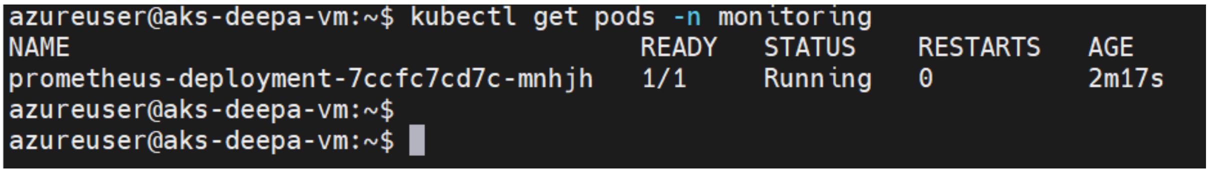


```
---
# ------------------------------------------------
# Prometheus Deployment
# ------------------------------------------------
apiVersion: apps/v1
kind: Deployment
metadata:
  name: prometheus-deployment
  namespace: monitoring
spec:
  replicas: 1
  selector:
    matchLabels:
      app: prometheus
  template:
    metadata:
      labels:
        app: prometheus
    spec:
      containers:
        - name: prometheus
          image: prom/prometheus:latest
          args:
            - "--config.file=/etc/prometheus/prometheus.yml"
          ports:
            - containerPort: 9090
          volumeMounts:
            - name: prometheus-config-volume
              mountPath: /etc/prometheus/
      volumes:
        - name: prometheus-config-volume
          configMap:
            name: prometheus-server-conf
---
```



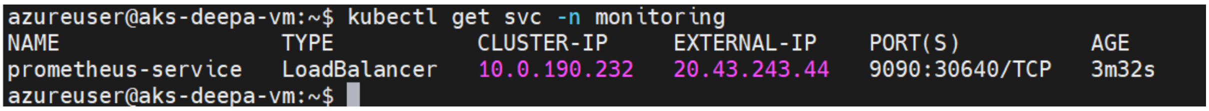
```
---
# ------------------------------------------------
# Prometheus Service (LoadBalancer)
# ------------------------------------------------
apiVersion: v1
kind: Service
metadata:
  name: prometheus-service
  namespace: monitoring
spec:
  type: LoadBalancer
  ports:
    - port: 9090
      targetPort: 9090
  selector:
    app: prometheus

azureuser@aks-deepa-vm:~$
```

Check deployment:



```
azureuser@aks-deepa-vm:~$ kubectl get pods -n monitoring
NAME                                    READY   STATUS    RESTARTS   AGE
prometheus-deployment-7ccfc7cd7c-mnhjh   1/1     Running   0          2m17s
azureuser@aks-deepa-vm:~$
azureuser@aks-deepa-vm:~$
```

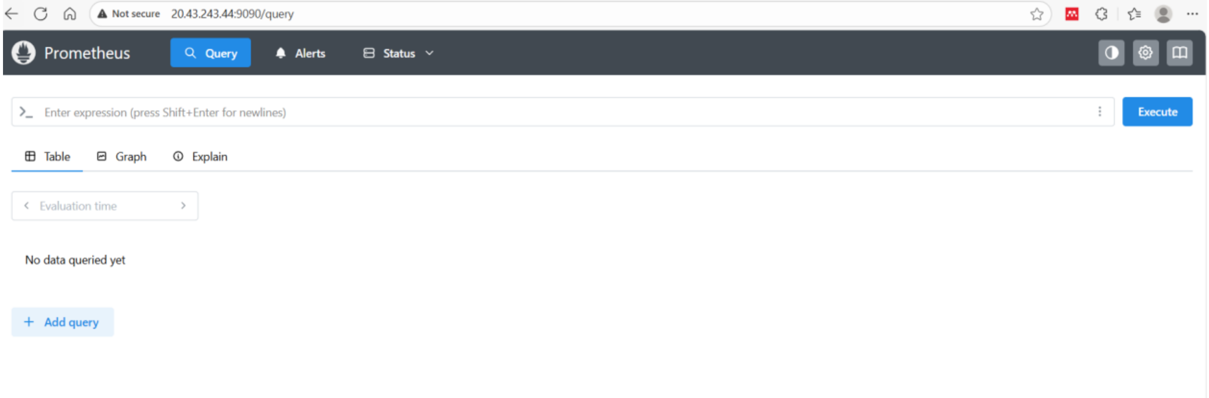
Check service and external IP:



```
azureuser@aks-deepa-vm:~$ kubectl get svc -n monitoring
NAME                 TYPE           CLUSTER-IP     EXTERNAL-IP    PORT(S)          AGE
prometheus-service   LoadBalancer   10.0.190.232   20.43.243.44   9090:30640/TCP   3m32s
azureuser@aks-deepa-vm:~$
```

http://20.43.243.44:9090

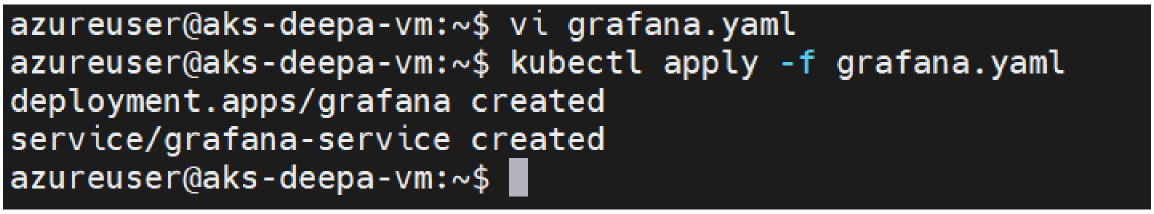
In browser



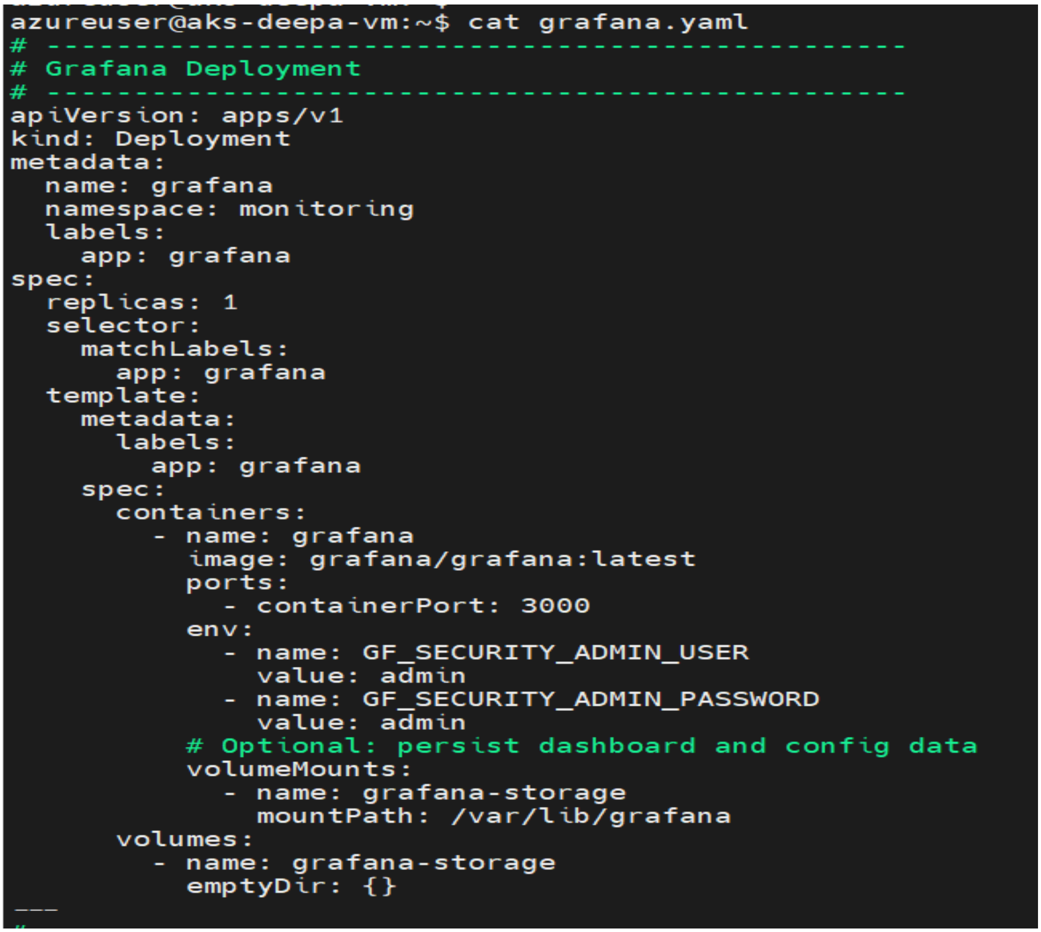
No data queried yet

To deploy Grafana:

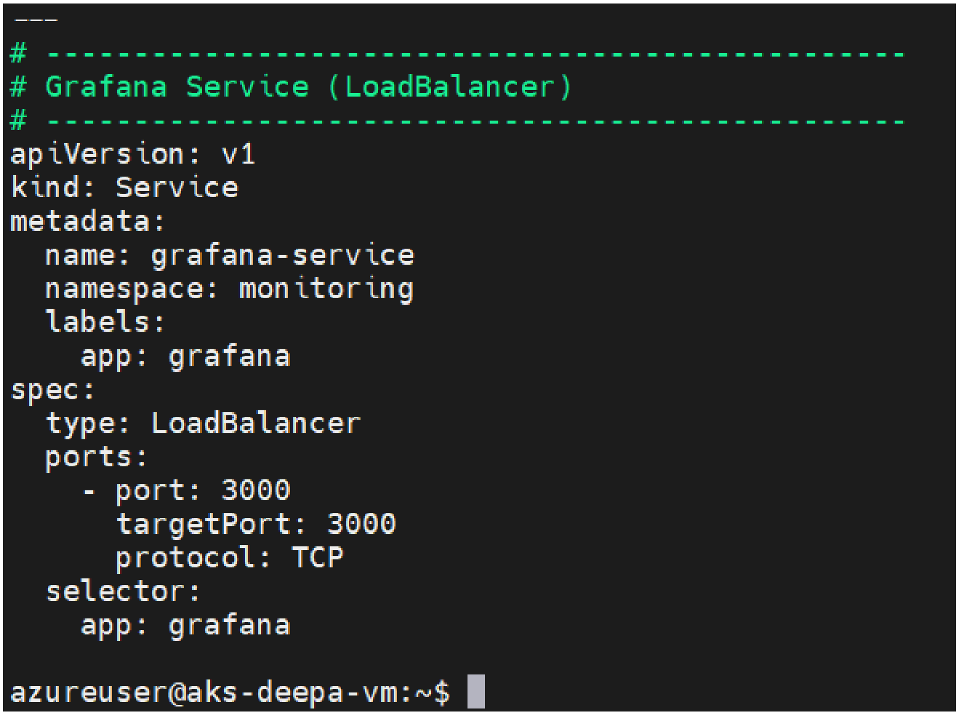
grafana.yaml



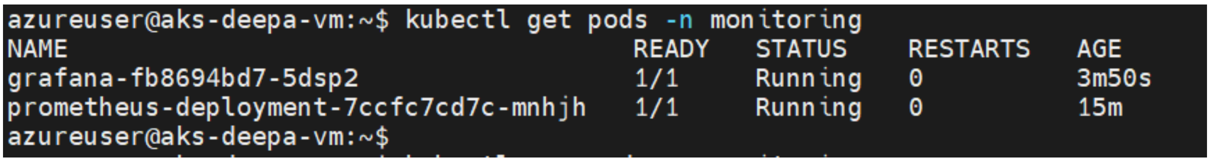
```
azureuser@aks-deepa-vm:~$ vi grafana.yaml
azureuser@aks-deepa-vm:~$ kubectl apply -f grafana.yaml
deployment.apps/grafana created
service/grafana-service created
azureuser@aks-deepa-vm:~$
```



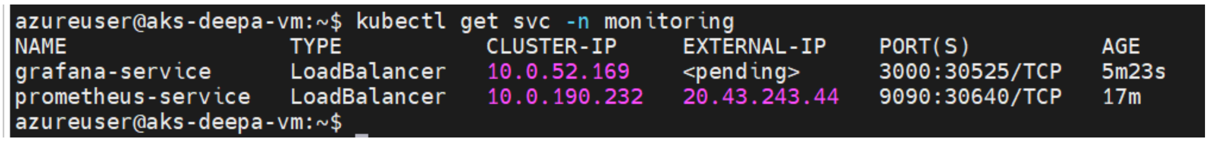
```
azureuser@aks-deepa-vm:~$ cat grafana.yaml
# --------------------------------------------------
# Grafana Deployment
# --------------------------------------------------
apiVersion: apps/v1
kind: Deployment
metadata:
  name: grafana
  namespace: monitoring
  labels:
    app: grafana
spec:
  replicas: 1
  selector:
    matchLabels:
      app: grafana
  template:
    metadata:
      labels:
        app: grafana
    spec:
      containers:
        - name: grafana
          image: grafana/grafana:latest
          ports:
            - containerPort: 3000
          env:
            - name: GF_SECURITY_ADMIN_USER
              value: admin
            - name: GF_SECURITY_ADMIN_PASSWORD
              value: admin
          # Optional: persist dashboard and config data
          volumeMounts:
            - name: grafana-storage
              mountPath: /var/lib/grafana
      volumes:
        - name: grafana-storage
          emptyDir: {}
---
```



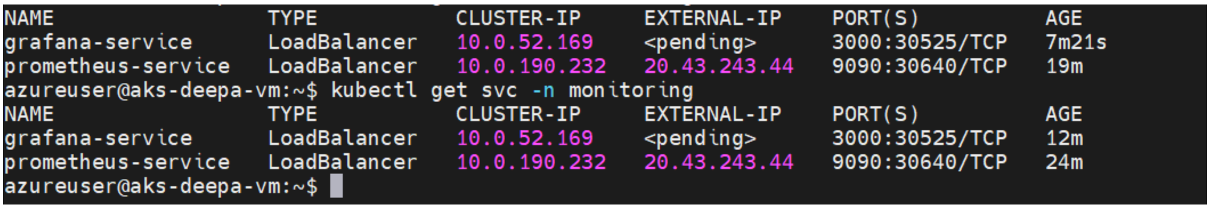
```
---
# --------------------------------------------------
# Grafana Service (LoadBalancer)
# --------------------------------------------------
apiVersion: v1
kind: Service
metadata:
  name: grafana-service
  namespace: monitoring
  labels:
    app: grafana
spec:
  type: LoadBalancer
  ports:
    - port: 3000
      targetPort: 3000
      protocol: TCP
  selector:
    app: grafana

azureuser@aks-deepa-vm:~$
```



```
azureuser@aks-deepa-vm:~$ kubectl get pods -n monitoring
NAME                                    READY   STATUS    RESTARTS   AGE
grafana-fb8694bd7-5dsp2                 1/1     Running   0          3m50s
prometheus-deployment-7ccfc7cd7c-mnhjh  1/1     Running   0          15m
azureuser@aks-deepa-vm:~$
```



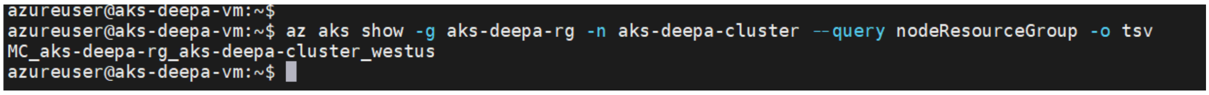
```
azureuser@aks-deepa-vm:~$ kubectl get svc -n monitoring
NAME                 TYPE           CLUSTER-IP     EXTERNAL-IP    PORT(S)          AGE
grafana-service      LoadBalancer   10.0.52.169    <pending>      3000:30525/TCP   5m23s
prometheus-service   LoadBalancer   10.0.190.232   20.43.243.44   9090:30640/TCP   17m
azureuser@aks-deepa-vm:~$ _
```



```
NAME                 TYPE           CLUSTER-IP     EXTERNAL-IP    PORT(S)          AGE
grafana-service      LoadBalancer   10.0.52.169    <pending>      3000:30525/TCP   7m21s
prometheus-service   LoadBalancer   10.0.190.232   20.43.243.44   9090:30640/TCP   19m
azureuser@aks-deepa-vm:~$ kubectl get svc -n monitoring
NAME                 TYPE           CLUSTER-IP     EXTERNAL-IP    PORT(S)          AGE
grafana-service      LoadBalancer   10.0.52.169    <pending>      3000:30525/TCP   12m
prometheus-service   LoadBalancer   10.0.190.232   20.43.243.44   9090:30640/TCP   24m
azureuser@aks-deepa-vm:~$
```

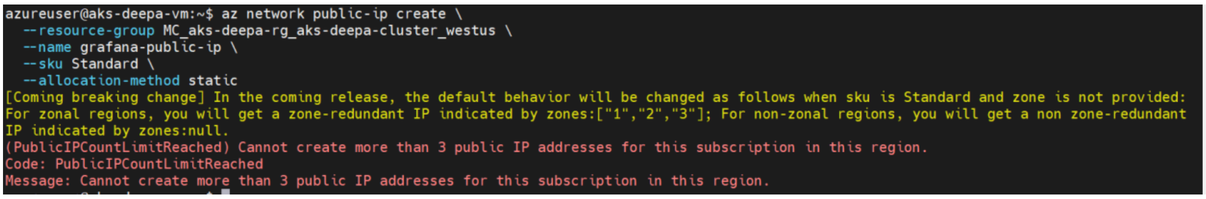
In AKS, **LoadBalancer Services** rely on the **Azure Load Balancer** integration.

Check AKS Node Resource Group



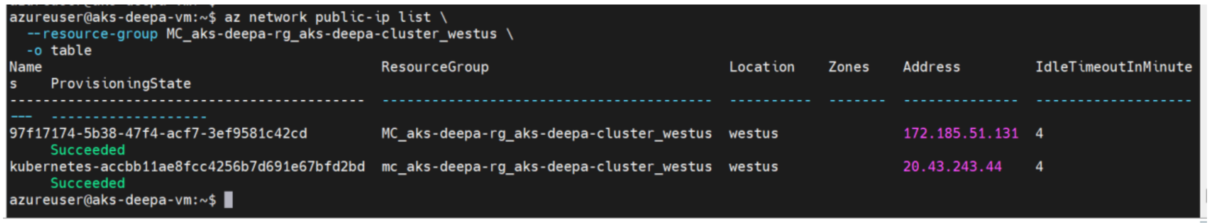
```
azureuser@aks-deepa-vm:~$
azureuser@aks-deepa-vm:~$ az aks show -g aks-deepa-rg -n aks-deepa-cluster --query nodeResourceGroup -o tsv
MC_aks-deepa-rg_aks-deepa-cluster_westus
azureuser@aks-deepa-vm:~$
```

Create a Public IP for Grafana:



```
azureuser@aks-deepa-vm:~$ az network public-ip create \
  --resource-group MC_aks-deepa-rg_aks-deepa-cluster_westus \
  --name grafana-public-ip \
  --sku Standard \
  --allocation-method static
[Coming breaking change] In the coming release, the default behavior will be changed as follows when sku is Standard and zone is not provided:
For zonal regions, you will get a zone-redundant IP indicated by zones:["1","2","3"]; For non-zonal regions, you will get a non zone-redundant
IP indicated by zones:null.
(PublicIPCountLimitReached) Cannot create more than 3 public IP addresses for this subscription in this region.
Code: PublicIPCountLimitReached
Message: Cannot create more than 3 public IP addresses for this subscription in this region.
```

List all public IPs in the node RG:



```
azureuser@aks-deepa-vm:~$ az network public-ip list \
  --resource-group MC_aks-deepa-rg_aks-deepa-cluster_westus \
  -o table
Name                                        ResourceGroup                             Location    Zones    Address         IdleTimeoutInMinute
s    ProvisioningState
------------------------------------------  ----------------------------------------  ----------  -------  --------------  -------------------
---  -------------------
97f17174-5b38-47f4-acf7-3ef9581c42cd        MC_aks-deepa-rg_aks-deepa-cluster_westus  westus               172.185.51.131  4
     Succeeded
kubernetes-accbb11ae8fcc4256b7d691e67bfd2bd  mc_aks-deepa-rg_aks-deepa-cluster_westus  westus               20.43.243.44    4
     Succeeded
azureuser@aks-deepa-vm:~$
```

Skip Public IP and Use Port-Forwarding (Quick Local Access)

Then open Grafana locally:

http://localhost:3000

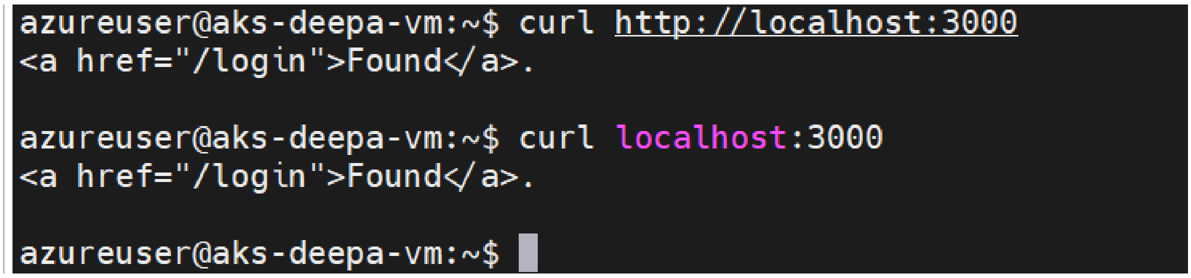
(Login: admin / admin)

This bypasses Azure load balancer limits.

Does **not** require a public IP or LoadBalancer.

Works even in a private or restricted cluster.

Closes automatically when we end the command (Ctrl+C).



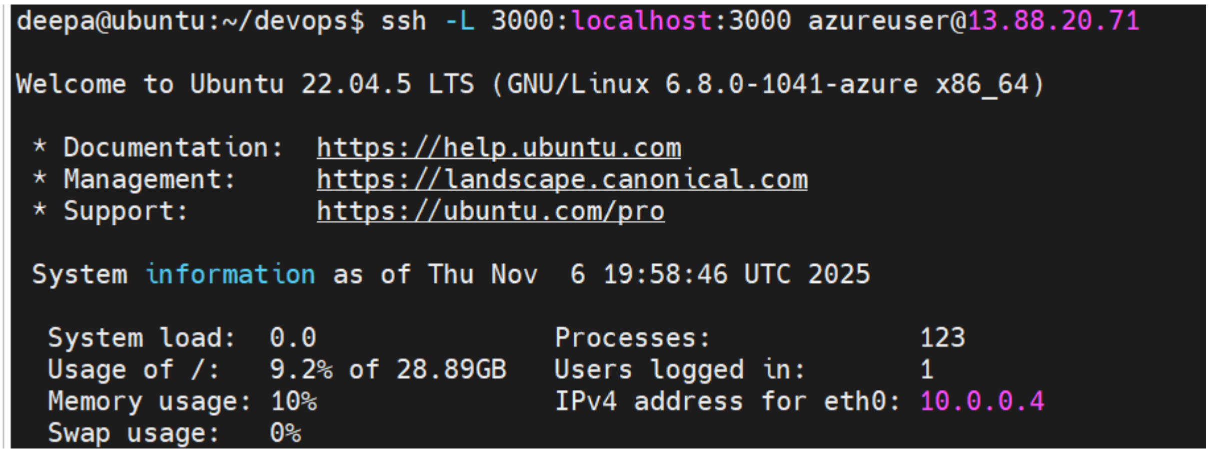
```
azureuser@aks-deepa-vm:~$ curl http://localhost:3000
<a href="/login">Found</a>.

azureuser@aks-deepa-vm:~$ curl localhost:3000
<a href="/login">Found</a>.

azureuser@aks-deepa-vm:~$
```

means Grafana **is running successfully** inside AKS cluster and reachable from **Azure VM** via port-forwarding.

So, the only reason browser on your laptop can't reach http://localhost:3000 is that **the port is being forwarded inside the VM, not on local computer**.

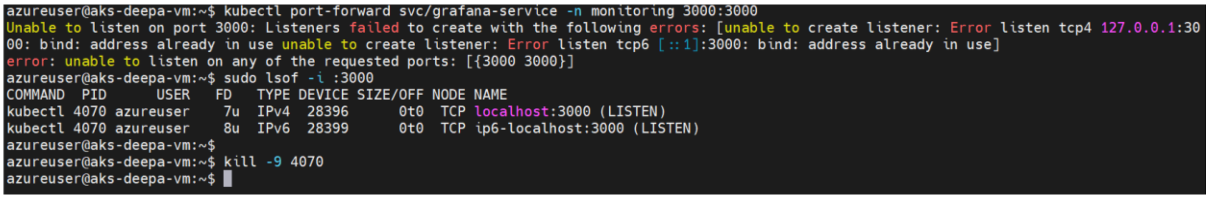
Open SSH Tunnel From Your Laptop:

In the local VM:



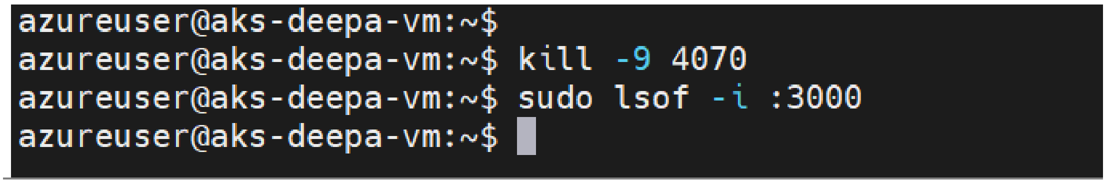
```
deepa@ubuntu:~/devops$ ssh -L 3000:localhost:3000 azureuser@13.88.20.71

Welcome to Ubuntu 22.04.5 LTS (GNU/Linux 6.8.0-1041-azure x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Thu Nov  6 19:58:46 UTC 2025

  System load:  0.0               Processes:             123
  Usage of /:   9.2% of 28.89GB   Users logged in:       1
  Memory usage: 10%               IPv4 address for eth0: 10.0.0.4
  Swap usage:   0%
```

Start Port-Forward for Grafana Inside the VM



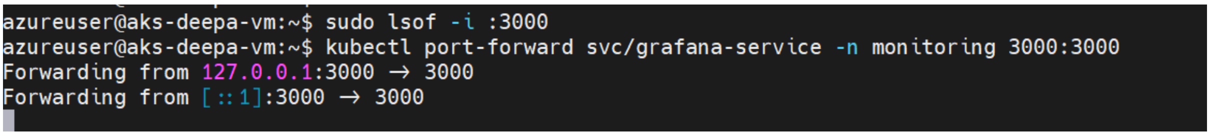
```
azureuser@aks-deepa-vm:~$ kubectl port-forward svc/grafana-service -n monitoring 3000:3000
Unable to listen on port 3000: Listeners failed to create with the following errors: [unable to create listener: Error listen tcp4 127.0.0.1:30
00: bind: address already in use unable to create listener: Error listen tcp6 [::1]:3000: bind: address already in use]
error: unable to listen on any of the requested ports: [{3000 3000}]
azureuser@aks-deepa-vm:~$ sudo lsof -i :3000
COMMAND  PID      USER   FD   TYPE DEVICE SIZE/OFF NODE NAME
kubectl 4070 azureuser    7u  IPv4  28396      0t0  TCP localhost:3000 (LISTEN)
kubectl 4070 azureuser    8u  IPv6  28399      0t0  TCP ip6-localhost:3000 (LISTEN)
azureuser@aks-deepa-vm:~$
azureuser@aks-deepa-vm:~$ kill -9 4070
azureuser@aks-deepa-vm:~$
```

After previous kubectl port-forward runs — they sometimes keep running in the background or didn't close properly.



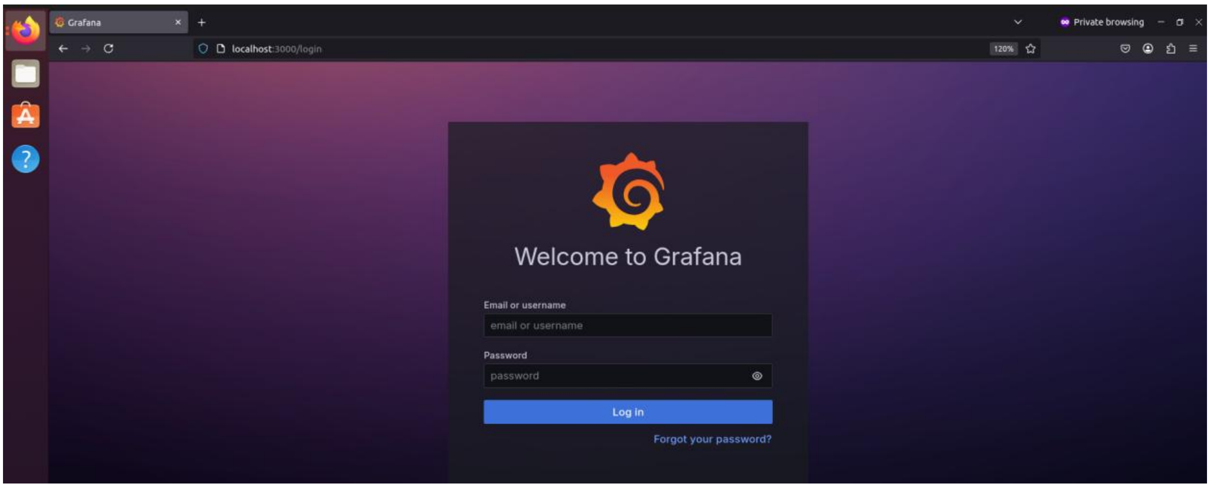
```
azureuser@aks-deepa-vm:~$
azureuser@aks-deepa-vm:~$ kill -9 4070
azureuser@aks-deepa-vm:~$ sudo lsof -i :3000
azureuser@aks-deepa-vm:~$
```

Now it is free



```
azureuser@aks-deepa-vm:~$ sudo lsof -i :3000
azureuser@aks-deepa-vm:~$ kubectl port-forward svc/grafana-service -n monitoring 3000:3000
Forwarding from 127.0.0.1:3000 -> 3000
Forwarding from [::1]:3000 -> 3000
```

http://localhost:3000



Welcome to Grafana

Email or username

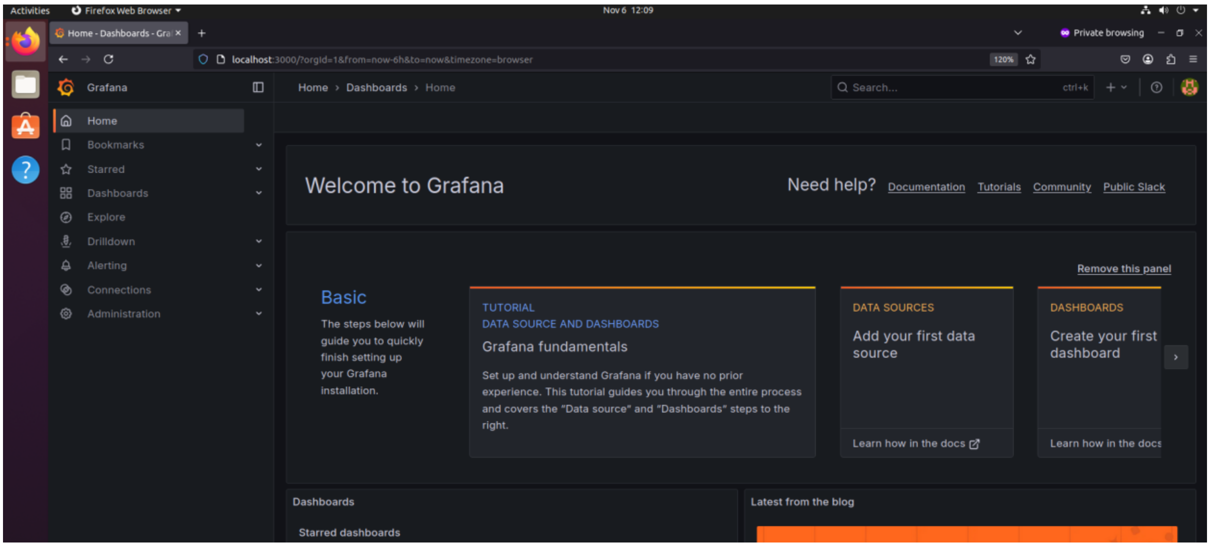
email or username

Password

password

Log in

Forgot your password?



Welcome to Grafana

Need help? Documentation Tutorials Community Public Slack

Remove this panel

Basic

The steps below will guide you to quickly finish setting up your Grafana installation.

TUTORIAL
DATA SOURCE AND DASHBOARDS
Grafana fundamentals

Set up and understand Grafana if you have no prior experience. This tutorial guides you through the entire process and covers the "Data source" and "Dashboards" steps to the right.

DATA SOURCES
Add your first data source
Learn how in the docs

DASHBOARDS
Create your first dashboard
Learn how in the docs

Dashboards

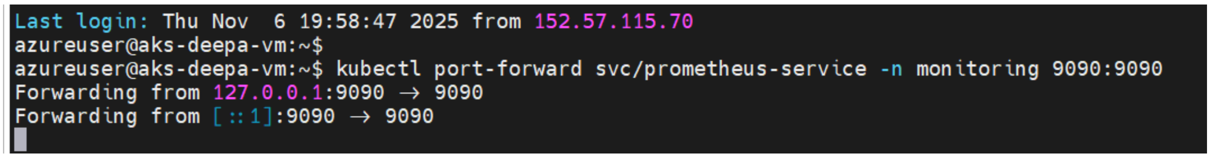
Starred dashboards

Latest from the blog

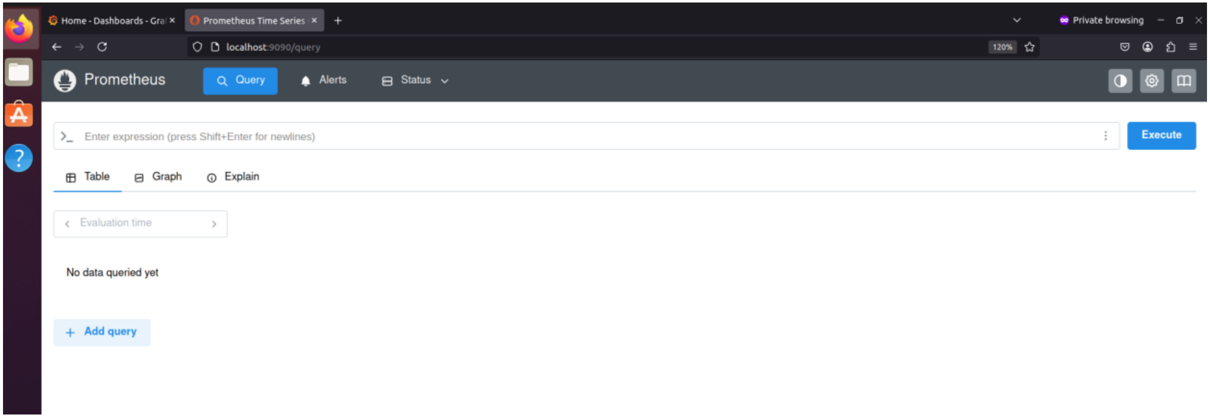
Open a Second SSH Session From Your Laptop

Keep the Grafana session open and open a **new terminal** on your **laptop**.

Inside the VM, Start Port-Forward for Prometheus



```
Last login: Thu Nov  6 19:58:47 2025 from 152.57.115.70
azureuser@aks-deepa-vm:~$
azureuser@aks-deepa-vm:~$ kubectl port-forward svc/prometheus-service -n monitoring 9090:9090
Forwarding from 127.0.0.1:9090 -> 9090
Forwarding from [::1]:9090 -> 9090
```



Enter expression (press Shift+Enter for newlines)

Execute

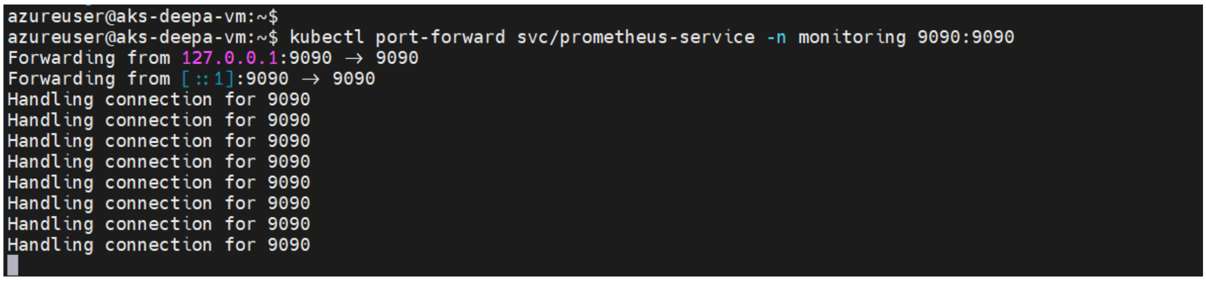
Evaluation time

No data queried yet

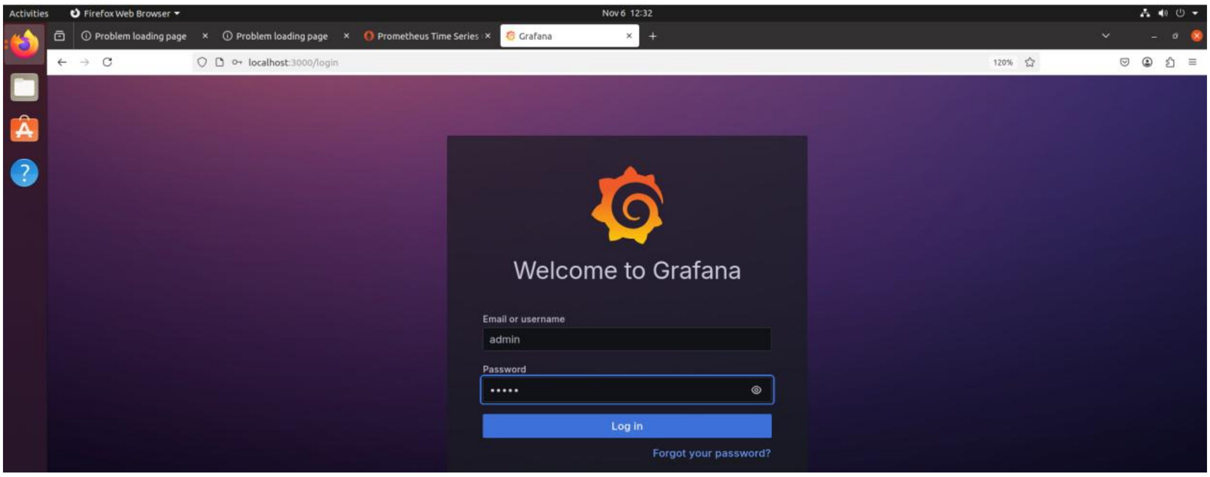
Add query

Inside the VM, Start Port-Forward for Prometheus

Inside the VM, Start Port-Forward for Prometheus



```
azureuser@aks-deepa-vm:~$
azureuser@aks-deepa-vm:~$ kubectl port-forward svc/prometheus-service -n monitoring 9090:9090
Forwarding from 127.0.0.1:9090 -> 9090
Forwarding from [::1]:9090 -> 9090
Handling connection for 9090
Handling connection for 9090
Handling connection for 9090
Handling connection for 9090
Handling connection for 9090
Handling connection for 9090
Handling connection for 9090
Handling connection for 9090
```



Welcome to Grafana

Email or username

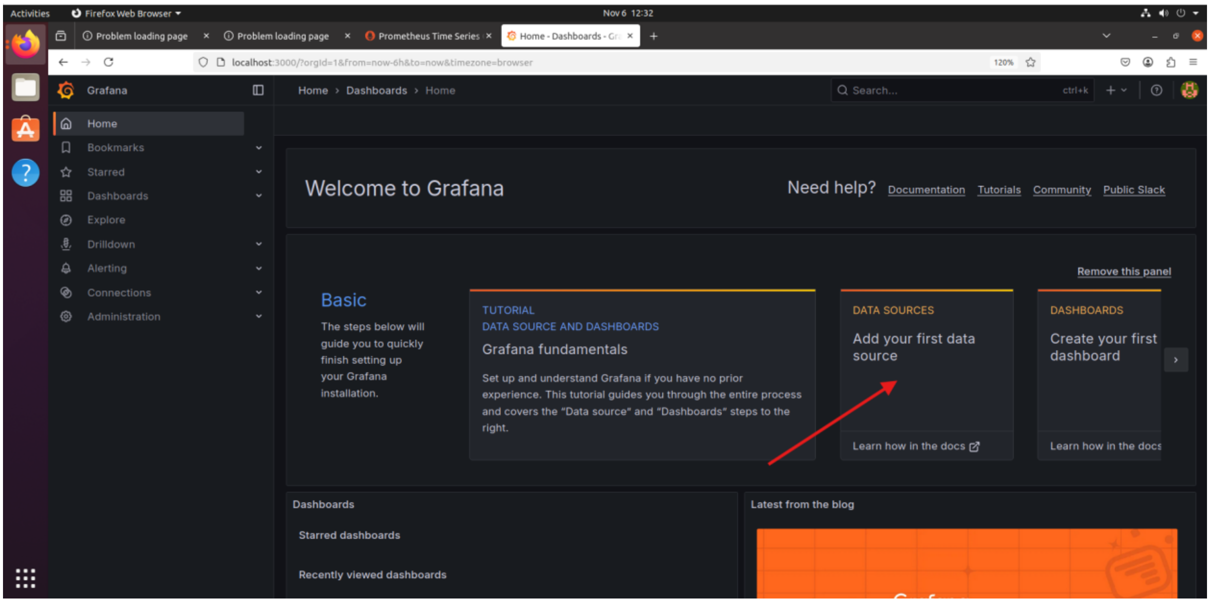
admin

Password

•••••

Log in

Forgot your password?



Welcome to Grafana

Need help? Documentation Tutorials Community Public Slack

Remove this panel

Basic

The steps below will guide you to quickly finish setting up your Grafana installation.

TUTORIAL
DATA SOURCE AND DASHBOARDS
Grafana fundamentals

Set up and understand Grafana if you have no prior experience. This tutorial guides you through the entire process and covers the "Data source" and "Dashboards" steps to the right.

DATA SOURCES
Add your first data source

Learn how in the docs

DASHBOARDS
Create your first dashboard

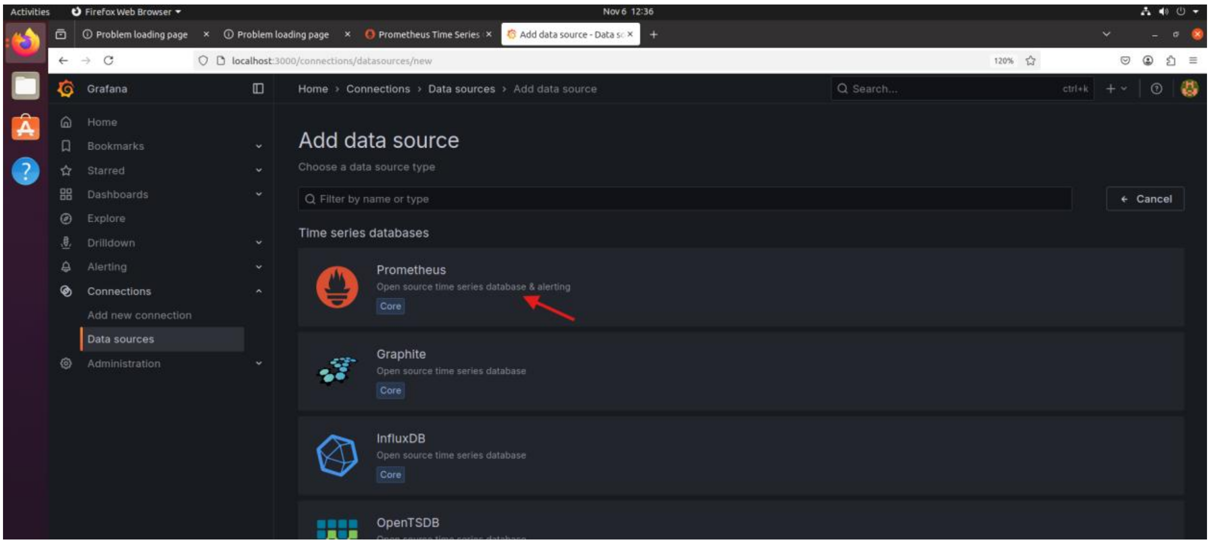
Learn how in the docs

Dashboards

Starred dashboards

Recently viewed dashboards

Latest from the blog



Add data source

Choose a data source type

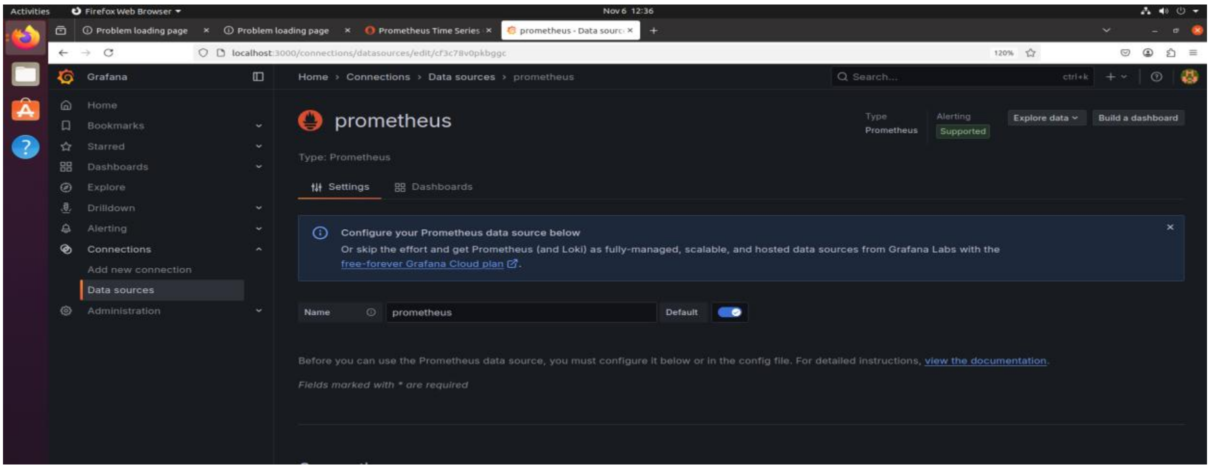
Time series databases

Prometheus
Open source time series database & alerting
Core

Graphite
Open source time series database
Core

InfluxDB
Open source time series database
Core

OpenTSDB



prometheus

Type: Prometheus

Type: Prometheus | Alerting: Supported

Configure your Prometheus data source below
Or skip the effort and get Prometheus (and Loki) as fully-managed, scalable, and hosted data sources from Grafana Labs with the free-forever Grafana Cloud plan.

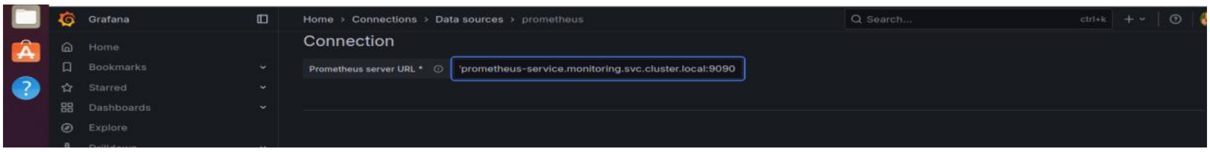
Name: prometheus | Default: on

Before you can use the Prometheus data source, you must configure it below or in the config file. For detailed instructions, view the documentation.

*Fields marked with * are required*

Home › Connections › Data sources › prometheus
Settings
Dashboards
Explore data
Build a dashboard
localhost:3000/connections/datasources/edit/cf3c78v0pkbggc

http://prometheus-service.monitoring.svc.cluster.local:9090

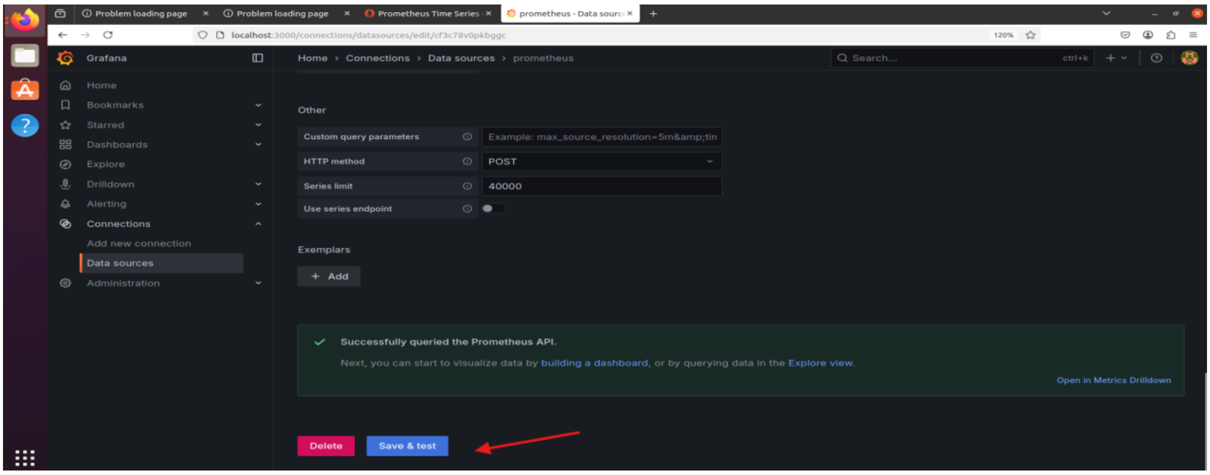


Connection

Prometheus server URL *: 'prometheus-service.monitoring.svc.cluster.local:9090

In Grafana:

- Click “+” → **Import**
- Enter Dashboard ID: **1860** (Node Exporter Full)
- Select your Prometheus data source → **Import**

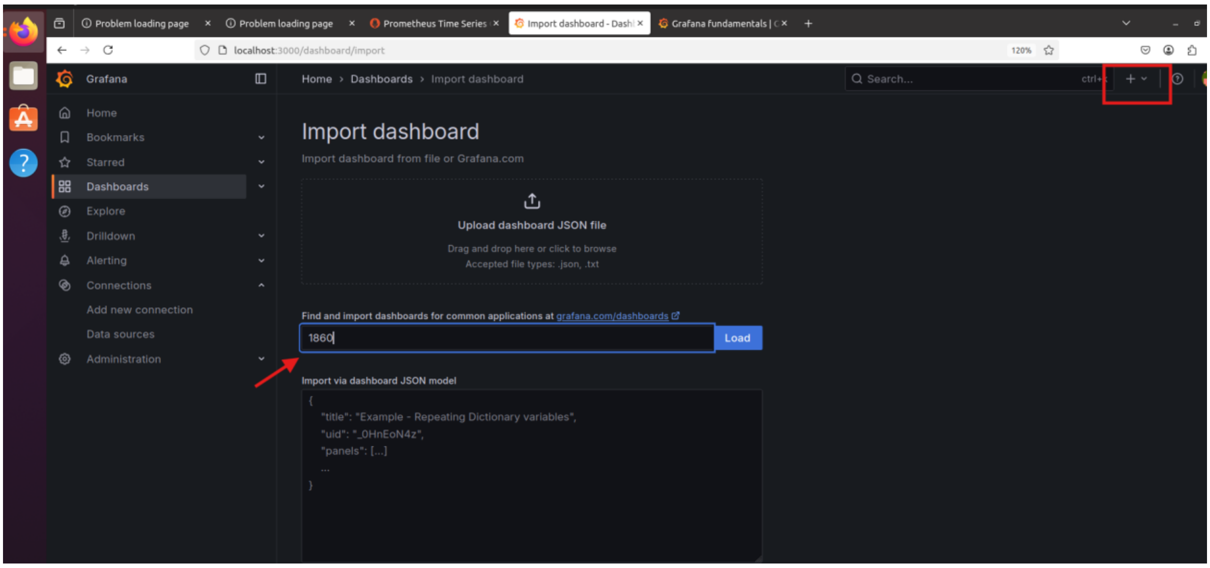


Other

Custom query parameters: Example: max_source_resolution=5m&amp;tin
HTTP method: POST
Series limit: 40000
Use series endpoint: off

Exemplars

Successfully queried the Prometheus API.
Next, you can start to visualize data by building a dashboard, or by querying data in the Explore view.



Import dashboard

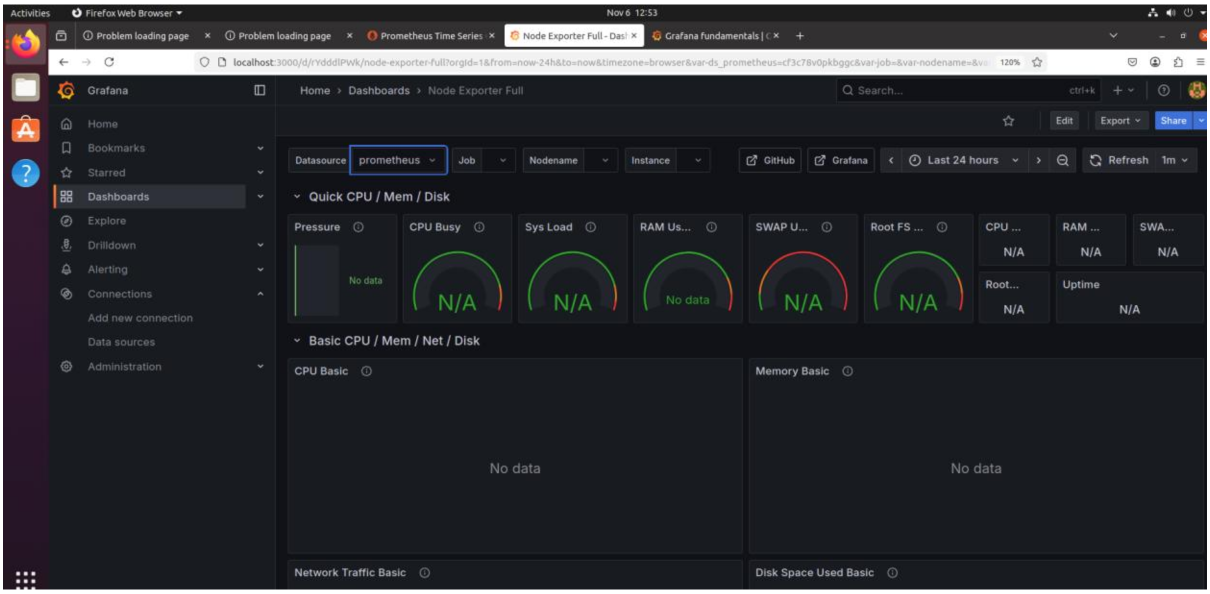
Import dashboard from file or Grafana.com

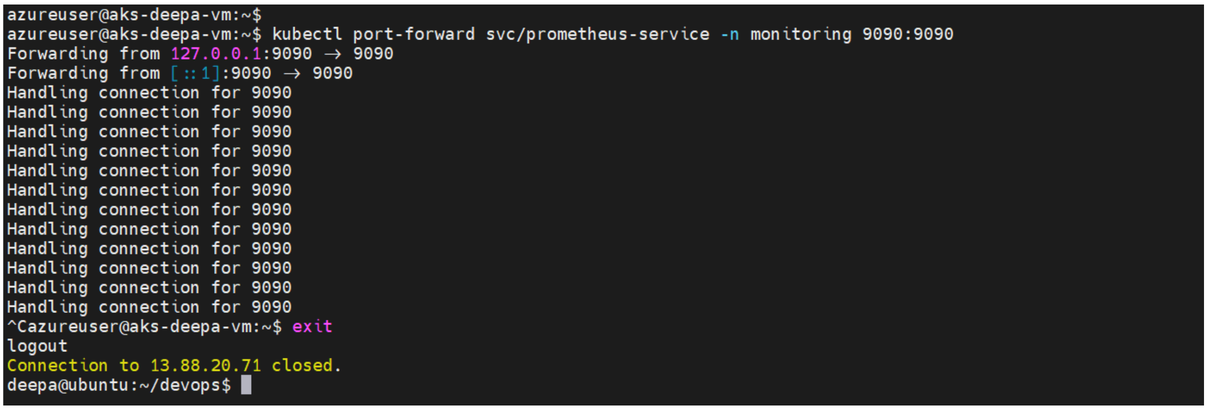
Find and import dashboards for common applications at grafana.com/dashboards

1860

Import via dashboard JSON model

```
{
  "title": "Example - Repeating Dictionary variables",
  "uid": "_0HnEoN4z",
  "panels": [...]
  ...
}
```





```
azureuser@aks-deepa-vm:~$
azureuser@aks-deepa-vm:~$ kubectl port-forward svc/prometheus-service -n monitoring 9090:9090
Forwarding from 127.0.0.1:9090 -> 9090
Forwarding from [::1]:9090 -> 9090
Handling connection for 9090
Handling connection for 9090
Handling connection for 9090
Handling connection for 9090
Handling connection for 9090
Handling connection for 9090
Handling connection for 9090
Handling connection for 9090
Handling connection for 9090
Handling connection for 9090
Handling connection for 9090
Handling connection for 9090
^Cazureuser@aks-deepa-vm:~$ exit
logout
Connection to 13.88.20.71 closed.
deepa@ubuntu:~/devops$ 
```