

Configure a Virtual Network for Azure Databricks (Private Preview)

Learn how to deploy Azure Databricks service to your own virtual network. This deployment enables the following scenarios:

- Connecting to Databricks directly from an on-premises network.
- Connecting Databricks to other Azure resources through service endpoints.

Note. This feature is in **private preview**. To request access, contact channel.sa@databricks.com.

[Workflow overview](#)

[Prepare a virtual network](#)

[Create a workspace with a custom virtual network](#)

[Create a workspace using the deployment template](#)

[Create a workspace on the Azure Portal](#)

[Appendix](#)

[Option 1 - All-in-one Template](#)

[Option 2 - Separate Templates](#)

[Network Security Group Template](#)

[Virtual Network Template](#)

[Databricks Workspace Template](#)

Workflow overview

1. Prepare a virtual network to which you would like to deploy a Databricks workspace. Two subnets are required in the virtual network:
 - a. Private subnet with an associated network security group that allows cluster-internal communication.
 - b. Public subnet with an associated network security group that allows communication with the Databricks control plane.
2. Create a Databricks workspace with the configured virtual network.

Prepare a virtual network

We recommend that you create a new virtual network to host each Databricks workspace, but you can use an existing virtual network if you prefer.

The virtual network requirements are as follows:

1. **Location:** The virtual network must be in the same region as the Databricks workspace you plan to create.
2. **Subnets:** Two subnets are required to host clusters:
 - a. Subnet named “Private”, with an associated network security group that allows cluster-internal communication.
 - b. Subnet named “Public”, with an associated network security group that allows communication with the Databricks control plane.

Certain network security rules must be associated with the subnets (covered in next section). It is recommended to create dedicated subnets for Databricks workspace, but you could also use existing subnets.

3. **Address space:** A /16 CIDR block is required for the virtual network. A /18 CIDR block is required for the private and public subnets.

To ensure proper cluster setup and communication, you must whitelist the following traffic on the subnets. It’s recommended to whitelist all outbound traffic because the Webapp IP could change.

Direction	Protocol	Source	Source Port	Destination	Destination Port
Inbound	*	VirtualNetwork	*	*	*
Inbound	*	Control Plane IP	*	*	22
Inbound	*	Control Plane IP	*	*	5557
Outbound	*	*	*	*	*

The control plane IP depends on the region:

Region	Control Plane IP
--------	------------------

australiaeast	13.70.105.50/32
australiasoutheast	13.70.105.50/32
centralus	23.101.152.95/32
eastasia	52.187.0.85/32
eastus	23.101.152.95/32
eastus2	23.101.152.95/32
northcentralus	23.101.152.95/32
northeurope	23.100.0.135/32
southcentralus	40.83.178.242/32
southeastasia	52.187.0.85/32
uksouth	51.140.203.27/32
ukwest	51.140.203.27/32
westeurope	23.100.0.135/32
westus	40.83.178.242/32
westus2	40.83.178.242/32

You can use the attached [template](#) to deploy a network security group that meets the above requirements, and then use the network [template](#) to deploy a virtual network with subnets protected by the network security group. This will give you a new virtual network that's ready for Databricks workspace creation.

Create a workspace with a custom virtual network

Three parameters are required in order to create a Databricks workspace with the custom virtual network.

Name	Type	Description
customVirtualNetworkId	String	Resource ID of the custom virtual network, e.g., "/subscriptions/36f75872-9ace-4c20-911c-aea8eba2945c/resourceGroups/tianyi-dev/providers/Microsoft.Network/virtualNetworks/tianyi-dev-vnet".
customPrivateSubnetName	String	Name of the private subnet in the custom virtual network, e.g., "private-subnet".

customPublicSubnetName	String	Name of the public subnet in the custom virtual network, e.g., "public-subnet".
------------------------	--------	---

Create a workspace using the deployment template

You can use the attached [template](#) to deploy the Databricks workspace.

Create a workspace on the Azure Portal

You can create the Databricks workspace using deployment template on Azure Portal by following these steps.

1. In the **Azure Portal**, [create an Azure Databricks service](#). On the Azure Databricks Service form, click **Automation options**.

Azure Databricks Service

* Workspace name
test ✓

* Subscription
Pay-As-You-Go

* Resource group ⓘ
☐ Create new ☒ Use existing
test

* Location
East US

* Pricing Tier ([View full pricing details](#))
Standard (Apache Spark, Secure with Azur... ▾

☐ Pin to dashboard

Create [Automation options](#)

2. Click **Deploy**.

Download Add to library Deploy

Automate deploying resources with Azure Resource Manager templates in a single, coordinated operation. Define resources and configurable input parameters and deploy with script or code. [Learn more about template deployment.](#)

Template Parameters CLI PowerShell .NET Ruby

Parameters (3)
Variables (2)
Resources (1)
[parameters('workspaceName')] (...)


```
1 {  
2   "$schema": "https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",  
3   "contentVersion": "1.0.0.0",  
4   "parameters": {  
5     "location": {  
6       "type": "string"  
7     },  
8     "workspaceName": {  
9       "type": "string"  
10    },  
11    "tier": {  
12      "defaultValue": "premium",  
13      "type": "string"  
14    }  
15  },  
16  "resources": [  
17    {  
18      "apiVersion": "2018-04-01",  
19      "location": "[parameters('location')]",  
20      "name": "[parameters('workspaceName')]",  
21      "sku": {  
22        "name": "[parameters('tier')]"  
23      },  
24      "comments": "Please do not use an existing resource group for ManagedResourceGroupId.",  
25      "properties": {  
26        "ManagedResourceGroupId": "[variables('managedResourceGroupId')]"  
27      },  
28      "type": "Microsoft.Databricks/workspaces"  
29    }  
30  ],  
31  "variables": {  
32    "managedResourceGroupId": "[concat(subscription().id, '/resourceGroups/', variables  
('managedResourceGroupName'))]",
```

3. On the **Custom deployment** page, click **Edit template**.

Custom deployment

Deploy from a custom template

TEMPLATE

 1 resource

Edit template

Edit parameters

Learn more

BASICS

* Subscription

Pay-As-You-Go

* Resource group

Create new

Use existing

Create a resource group

* Location

East US

SETTINGS

* Location

eastus

* Workspace Name

test

Tier

standard

TERMS AND CONDITIONS

[Azure Marketplace Terms](#)

[Azure Marketplace](#)

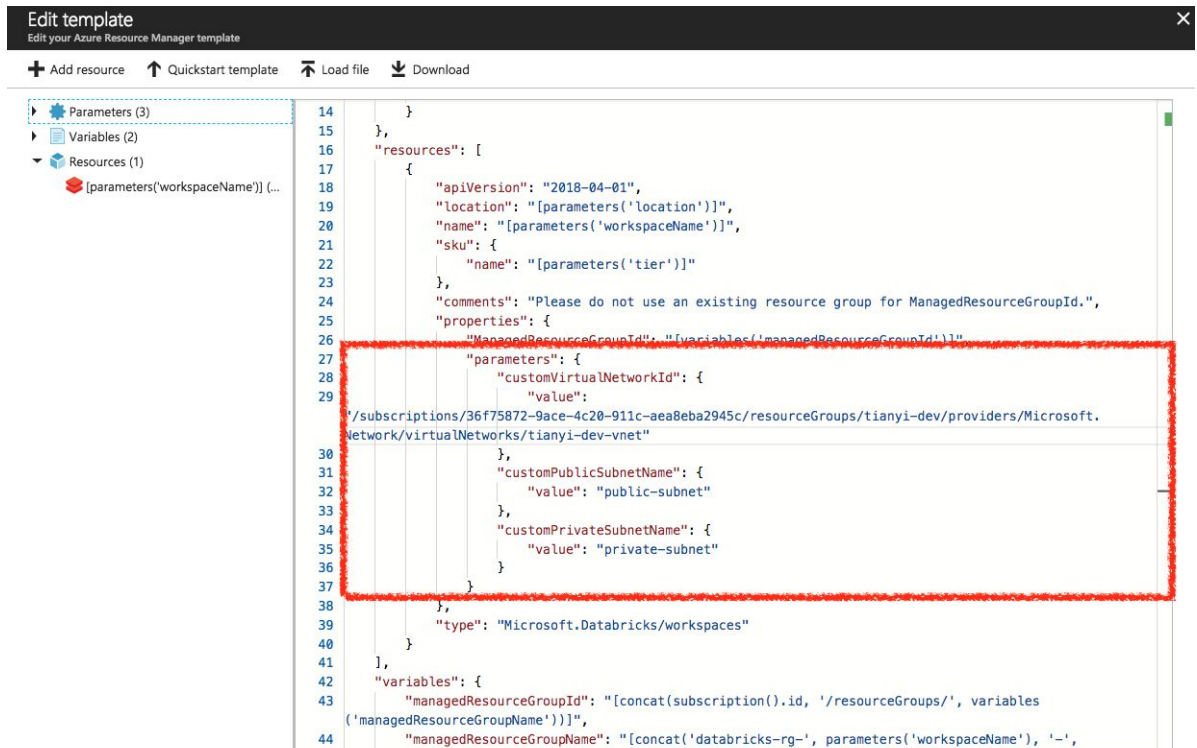
By clicking "Purchase," I (a) agree to the applicable legal terms associated with the offering; (b) authorize Microsoft to charge or bill my current payment method for the fees associated the offering(s), including applicable taxes, with the same billing frequency as my Azure subscription, until I discontinue use of the offering(s); and (c) agree that, if the deployment involves 3rd party offerings, Microsoft may share my contact information and other details of such deployment with the publisher of that offering.

Microsoft assumes no responsibility for any actions performed by third-party templates and does not provide rights for third-party

☐ Pin to dashboard

Purchase

4. Edit the template to include the custom virtual network parameters.



5. Click **Save**.

6. Fill out the required fields and click **Purchase**.

Appendix

Option 1 - All-in-one Template

Deploy using Azure CLI 2.0

```
az group deployment create --name $deployment --resource-group
$resourceGroup --template-file $file --parameters nsgName=$nsgName
vnetName=$vnetName vnetAddressPrefix=$vnetAddressPrefix
workspaceName=$workspaceName
```

```
{
  "$schema":
    "https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",
  "contentVersion": "1.0.0.0",
  "parameters": {
    "nsgName": {
      "defaultValue": "databricks-nsg",
      "type": "string"
    },
  },
}
```

```

    "vnetName": {
      "defaultValue": "databricks-vnet",
      "type": "string"
    },
    "workspaceName": {
      "type": "string"
    },
    "vnetAddressPrefix": {
      "defaultValue": "10.179",
      "type": "string"
    },
    "privateSubnetName": {
      "defaultValue": "private-subnet",
      "type": "string"
    },
    "publicSubnetName": {
      "defaultValue": "public-subnet",
      "type": "string"
    },
    "workspaceTier": {
      "defaultValue": "premium",
      "type": "string"
    }
  },
  "resources": [
    {
      "apiVersion": "2018-02-01",
      "type": "Microsoft.Network/networkSecurityGroups",
      "location": "[resourceGroup().location]",
      "name": "[parameters('nsgName')]",
      "properties": {
        "securityRules": [
          {
            "name": "databricks-worker-to-worker",
            "properties": {
              "access": "Allow",
              "description": "Required for worker nodes communication within
a cluster.",
              "destinationAddressPrefix": "*",
              "destinationPortRange": "*",
              "direction": "Inbound",
              "priority": 200,
              "protocol": "*",
              "sourceAddressPrefix": "VirtualNetwork",
              "sourcePortRange": "*"
            }
          },
          {
            "name": "databricks-control-plane-ssh",
            "properties": {
              "access": "Allow",
              "description": "Required for Databricks control plane
management of worker nodes.",
              "destinationAddressPrefix": "*",
              "destinationPortRange": "22",
              "direction": "Inbound",
              "priority": 100,
              "protocol": "*",
              "sourceAddressPrefix": "[variables('controlPlaneIp')]",
              "sourcePortRange": "*"
            }
          }
        ]
      }
    }
  ]
}

```



```

    }
  },
  {
    "name": "databricks-control-plane-worker-proxy",
    "properties": {
      "access": "Allow",
      "description": "Required for Databricks control plane
communication with worker nodes.",
      "destinationAddressPrefix": "*",
      "destinationPortRange": "5557",
      "direction": "Inbound",
      "priority": 110,
      "protocol": "*",
      "sourceAddressPrefix": "[variables('controlPlaneIp')]",
      "sourcePortRange": "*"
    }
  },
  {
    "name": "databricks-worker-to-internet",
    "properties": {
      "access": "Allow",
      "description": "Required for workers communication with
Internet.",
      "destinationAddressPrefix": "*",
      "destinationPortRange": "*",
      "direction": "Outbound",
      "priority": 120,
      "protocol": "*",
      "sourceAddressPrefix": "*",
      "sourcePortRange": "*"
    }
  }
]
}
},
{
  "apiVersion": "2018-02-01",
  "type": "Microsoft.Network/virtualNetworks",
  "location": "[resourceGroup().location]",
  "name": "[parameters('vnetName')]",
  "dependsOn": [
    "[concat('Microsoft.Network/networkSecurityGroups/',
parameters('nsgName'))]"
  ],
  "properties": {
    "addressSpace": {
      "addressPrefixes": [
        "[variables('vnetCidr')]"
      ]
    },
    "subnets": [
      {
        "name": "[parameters('publicSubnetName')]",
        "properties": {
          "addressPrefix": "[variables('publicSubnetCidr')]",
          "networkSecurityGroup": {
            "id": "[variables('nsgId')]"
          }
        }
      }
    ]
  },

```

```

        {
            "name": "[parameters('privateSubnetName')]",
            "properties": {
                "addressPrefix": "[variables('privateSubnetCidr')]",
                "networkSecurityGroup": {
                    "id": "[variables('nsgId')]"
                }
            }
        }
    ]
}
},
{
    "apiVersion": "2018-04-01",
    "type": "Microsoft.Databricks/workspaces",
    "location": "[resourceGroup().location]",
    "name": "[parameters('workspaceName')]",
    "dependsOn": [
        "[concat('Microsoft.Network/networkSecurityGroups/',
parameters('nsgName'))]",
        "[concat('Microsoft.Network/virtualNetworks/', parameters('vnetName'))]"
    ],
    "sku": {
        "name": "[parameters('workspaceTier')]"
    },
    "comments": "Please do not use an existing resource group for
ManagedResourceId.",
    "properties": {
        "ManagedResourceId": "[variables('managedResourceId')]",
        "parameters": {
            "customVirtualNetworkId": {
                "value": "[variables('vnetId')]",
            },
            "customPublicSubnetName": {
                "value": "[parameters('publicSubnetName')]",
            },
            "customPrivateSubnetName": {
                "value": "[parameters('privateSubnetName')]",
            }
        }
    }
}
},
{
    "variables": {
        "azureRegionToControlPlaneIp": {
            "australiaeast": "13.70.105.50/32",
            "australiasoutheast": "13.70.105.50/32",
            "centralus": "23.101.152.95/32",
            "eastasia": "52.187.0.85/32",
            "eastus": "23.101.152.95/32",
            "eastus2": "23.101.152.95/32",
            "northcentralus": "23.101.152.95/32",
            "northeurope": "23.100.0.135/32",
            "southcentralus": "40.83.178.242/32",
            "southeastasia": "52.187.0.85/32",
            "uksouth": "51.140.203.27/32",
            "ukwest": "51.140.203.27/32",
            "westeurope": "23.100.0.135/32",
            "westus": "40.83.178.242/32",
            "westus2": "40.83.178.242/32"
        }
    }
}

```

```

    },
    "controlPlaneIp":
"[variables('azureRegionToControlPlaneIp')[resourceGroup().location]]",
    "privateSubnetCidr": "[concat(parameters('vnetAddressPrefix'), '.64.0/18')]",
    "publicSubnetCidr": "[concat(parameters('vnetAddressPrefix'), '.0.0/18')]",
    "vnetCidr": "[concat(parameters('vnetAddressPrefix'), '.0.0/16')]",
    "nsgId": "[resourceId('Microsoft.Network/networkSecurityGroups',
parameters('nsgName'))]",
    "vnetId": "[resourceId('Microsoft.Network/virtualNetworks',
parameters('vnetName'))]",
    "managedResourceId": "[concat(subscription().id, '/resourceGroups/',
variables('managedResourceGroupName'))]",
    "managedResourceGroupName": "[concat('databricks-rg-',
parameters('workspaceName'), '-', uniqueString(parameters('workspaceName'),
resourceGroup().id))]"
  }
}

```

Option 2 - Separate Templates

Network Security Group Template

Deploy using Azure CLI 2.0

```

az group deployment create --name $deployment --resource-group
$resourceGroup --template-file $file --parameters nsgName=$nsgName

```

```

{
  "$schema":
"https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",
  "contentVersion": "1.0.0.0",
  "parameters": {
    "nsgName": {
      "defaultValue": "databricks-nsg",
      "type": "string"
    }
  },
  "resources": [
    {
      "apiVersion": "2018-02-01",
      "location": "[resourceGroup().location]",
      "name": "[parameters('nsgName')]",
      "properties": {
        "securityRules": [
          {
            "name": "databricks-worker-to-worker",
            "properties": {
              "access": "Allow",
              "description": "Required for worker nodes communication within
a cluster.",
              "destinationAddressPrefix": "*",
              "destinationPortRange": "*",
              "direction": "Inbound",
              "priority": 200,

```

```

        "protocol": "*",
        "sourceAddressPrefix": "VirtualNetwork",
        "sourcePortRange": "*"
    }
},
{
    "name": "databricks-control-plane-ssh",
    "properties": {
        "access": "Allow",
        "description": "Required for Databricks control plane
management of worker nodes.",
        "destinationAddressPrefix": "*",
        "destinationPortRange": "22",
        "direction": "Inbound",
        "priority": 100,
        "protocol": "*",
        "sourceAddressPrefix": "[variables('controlPlaneIp')]",
        "sourcePortRange": "*"
    }
},
{
    "name": "databricks-control-plane-worker-proxy",
    "properties": {
        "access": "Allow",
        "description": "Required for Databricks control plane
communication with worker nodes.",
        "destinationAddressPrefix": "*",
        "destinationPortRange": "5557",
        "direction": "Inbound",
        "priority": 110,
        "protocol": "*",
        "sourceAddressPrefix": "[variables('controlPlaneIp')]",
        "sourcePortRange": "*"
    }
},
{
    "name": "databricks-worker-to-internet",
    "properties": {
        "access": "Allow",
        "description": "Required for workers communication with
Internet.",
        "destinationAddressPrefix": "*",
        "destinationPortRange": "*",
        "direction": "Outbound",
        "priority": 120,
        "protocol": "*",
        "sourceAddressPrefix": "*",
        "sourcePortRange": "*"
    }
}
    ]
},
    "type": "Microsoft.Network/networkSecurityGroups"
}
],
"variables": {
    "azureRegionToControlPlaneIp": {
        "australiaeast": "13.70.105.50/32",
        "australiasoutheast": "13.70.105.50/32",
        "centralus": "23.101.152.95/32",

```

```

        "eastasia": "52.187.0.85/32",
        "eastus": "23.101.152.95/32",
        "eastus2": "23.101.152.95/32",
        "northcentralus": "23.101.152.95/32",
        "northeurope": "23.100.0.135/32",
        "southcentralus": "40.83.178.242/32",
        "southeastasia": "52.187.0.85/32",
        "uksouth": "51.140.203.27/32",
        "ukwest": "51.140.203.27/32",
        "westeurope": "23.100.0.135/32",
        "westus": "40.83.178.242/32",
        "westus2": "40.83.178.242/32"
    },
    "controlPlaneIp":
    "[variables('azureRegionToControlPlaneIp')[resourceGroup().location]]"
  }
}

```

Virtual Network Template

Deploy using Azure CLI 2.0

```

az group deployment create --name $deployment --resource-group
$resourceGroup --template-file $file --parameters vnetName=$vnetName
nsgId=$nsgId

```

```

{
  "$schema":
  "https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",
  "contentVersion": "1.0.0.0",
  "parameters": {
    "vnetName": {
      "defaultValue": "databricks-vnet",
      "type": "string"
    },
    "vnetAddressPrefix": {
      "defaultValue": "10.179",
      "type": "string"
    },
    "privateSubnetName": {
      "defaultValue": "private-subnet",
      "type": "string"
    },
    "publicSubnetName": {
      "defaultValue": "public-subnet",
      "type": "string"
    },
    "nsgId": {
      "defaultValue": "",
      "type": "string"
    }
  },
  "resources": [
    {
      "apiVersion": "2018-02-01",
      "location": "[resourceGroup().location]",

```

```

        "name": "[parameters('vnetName')]",
        "properties": {
            "addressSpace": {
                "addressPrefixes": [
                    "[variables('vnetCidr')]"
                ]
            },
            "subnets": [
                {
                    "name": "[parameters('publicSubnetName')]",
                    "properties": {
                        "addressPrefix": "[variables('publicSubnetCidr')]",
                        "networkSecurityGroup": {
                            "id": "[parameters('nsgId')]"
                        }
                    }
                },
                {
                    "name": "[parameters('privateSubnetName')]",
                    "properties": {
                        "addressPrefix": "[variables('privateSubnetCidr')]",
                        "networkSecurityGroup": {
                            "id": "[parameters('nsgId')]"
                        }
                    }
                }
            ]
        },
        "type": "Microsoft.Network/virtualNetworks"
    },
    "variables": {
        "privateSubnetCidr": "[concat(parameters('vnetAddressPrefix'), '.64.0/18')]",
        "publicSubnetCidr": "[concat(parameters('vnetAddressPrefix'), '.0.0/18')]",
        "vnetCidr": "[concat(parameters('vnetAddressPrefix'), '.0.0/16')]"
    }
}

```

Databricks Workspace Template

Deploy using Azure CLI 2.0

```

az group deployment create --name $deployment --resource-group
$resourceGroup --template-file $file --parameters
workspaceName=$workspaceName customVirtualNetworkId=$vnetId

```

```

{
    "$schema":
    "https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",
    "contentVersion": "1.0.0.0",
    "parameters": {
        "workspaceName": {
            "type": "string"
        },
        "tier": {
            "defaultValue": "premium",

```

```

        "type": "string"
    },
    "customVirtualNetworkId": {
        "type": "string",
        "defaultValue": "",
    },
    "customPublicSubnetName": {
        "type": "string",
        "defaultValue": "public-subnet",
    },
    "customPrivateSubnetName": {
        "type": "string",
        "defaultValue": "private-subnet",
    }
},
"resources": [
    {
        "apiVersion": "2018-04-01",
        "location": "[resourceGroup().location]",
        "name": "[parameters('workspaceName')]",
        "sku": {
            "name": "[parameters('tier')]"
        },
        "comments": "Please do not use an existing resource group for
ManagedResourceGroupId.",
        "properties": {
            "ManagedResourceGroupId": "[variables('managedResourceGroupId')]",
            "parameters": {
                "customVirtualNetworkId": {
                    "value": "[parameters('customVirtualNetworkId')]",
                },
                "customPublicSubnetName": {
                    "value": "[parameters('customPublicSubnetName')]",
                },
                "customPrivateSubnetName": {
                    "value": "[parameters('customPrivateSubnetName')]",
                }
            }
        },
        "type": "Microsoft.Databricks/workspaces"
    }
],
"variables": {
    "managedResourceGroupId": "[concat(subscription().id, '/resourceGroups/',
variables('managedResourceGroupName'))]",
    "managedResourceGroupName": "[concat('databricks-rg-',
parameters('workspaceName'), '-', uniqueString(parameters('workspaceName'),
resourceGroup().id))]"
}
}

```