**Azure Databricks Deployment Process**

**Resource Groups:**

Databricks has some provision’s restrictions that sometimes throw Errors if they are not prepare for them.

When deploying Databricks you deploy a Databricks workspace. A workspace is an Azure resource of type Microsoft.Databricks/workspaces. This is the only resource being deployed in the resource group that you choose to contain the workspace.

**Managed Resource Group:**

Every Databricks workspace has a resource group managed by it and contains the resources that the workspace needs. Initially the group will contain

* Storage Account (that is your dbfs)
* Virtual Network (if choose not to have vnet integration)
* Network Security Group (NSG) (if choose not to have vnet integration)

The group is locked by a read-only lock by the Databricks service principal which you cannot delete yourself even if you are the owner of the subscription.

The name of the resource group by default will be

databricks-rg-<workspace Name>-<random string>

You can see which resource group is managed by your workspace when you navigate to the workspace overview blade.

We cannot combine multiple workspaces in the same managed Group one workspace, one managed resource group.

We cannot pre-create the resource group before creating the workspace as the managed resource group has to be created by the workspace.

**Virtual Networks & Clusters:**

Azure Databricks **workspace** is a code authoring and collaboration workspace that can have one or more Apache Spark clusters. So as a prerequisites to create the cluster, there has to be a **Virtual Network (VNET)** to have the machines in it. That is why in the managed resource group, if you do not choose to use your own VNET, there will be a VNET created for you.

Organizations have restrictions about the VNET creation and prefer to integrate the Databricks clusters into their network infrastructure, that’s why Azure Databricks now supports VNET injection. VNET injection enable you to use an existing VNET for hosting the Databricks clusters.

The requirements of these VNETs

* Have to be in the same subscription
* **Address space:**  virtual network, private and public subnets
* Two subnet per workspace
* Cannot use the same subnet by two workspaces

The control plane and web UI are always deployed in a Microsoft-managed subscription. That is why the Azure Databricks UI is always https://<Azure Region>.azuredatabricks.net

During workspace deployment, there is no clusters created yet. So, during deployment, Databricks would ensure that the minimum requirements for the clusters to be successfully deployed are met.

**During deployment time:**

* Delegation to Microsoft.Databricks/workspaces is made on each subnet. This delegation will be used by the service to create a Network Intent Policy. Azure Network Intent Policy is a feature used by some Microsoft first party providers like SQL Managed Instance. It is not available for customers’ use as there is no public docs for it. In the case of Databricks, it is used to maintain the NSG rules added by the workspace creation. If you try to delete one of the rules, you will get an error message because of the Intent Policy.
* Check if there is already an NSG attached to the subnet. If not, then the creation of the workspace will fail. It will create only one NSG for both private & public subnets. The rules are identical anyways even if you provided two NSGs, you would find the same set of rules.
* If there’s NSG attached, new rules will be added and protected by the Intent Policy so it cannot be changed or deleted.

**Two subnets:**

Each machine in the Databricks cluster has two virtual network cards (NIC), one with private IP only attached to the private subnet and one with both private & public IPs attached to public subnets. The public IPs used to communicate between your cluster and the control plane of Azure Databricks plus any other data sources that might reside outside your Vnet. All the inter-cluster communication happens on that.

**Integrating Azure Databricks workspaces with firewalls:**

One of the main reasons to have Azure Databricks workspaces integrated with VNET is to utilize your existing firewalls. The workflow to do that is

* Create and assign Route Tables that will force the traffic from Azure Databricks to be filtered by the firewall first.
* In the firewall you add rules to allow the traffic needed for your workspace

The traffic coming to the public IPs of the workspace clusters, doesn’t pass on the firewall and if your routing route all 0.0.0.0/0 to the firewall that means the firewall will only see the return of the traffic and for any stateful firewall like Azure Firewall, it will drop the traffic. You have to add exception of the routing for the control plane.