# azurerm_databricks_workspace

Manages a Databricks Workspace

## Example Usage

```
resource "azurerm_resource_grou
  name     = "example-resources"
  location = "West Europe"
}

resource "azurerm_databricks_workspace"
  name                = "databricks-test
  resource_group_name = azurerm_resource
  location            = azurerm_resource
  sku                 = "standard"

  tags = {
    Environment = "Production"
  }
}
```

Copy

**Note**

You can use the Databricks Terraform

## Argument Reference

The following arguments are supported:

- `name` - (Required) Specifies the name of the Databricks Workspace resource. Changing this forces a new resource to be created.

- `resource_group_name` - (Required) The name of the Resource Group in which the Databricks Workspace should exist. Changing this forces a new resource to be created.

- `location` - (Required) Specifies the supported Azure location where the resource has to be created. Changing this forces a new resource to be created.

- `load_balancer_backend_address_pool_id` - (Optional) Resource ID of the Outbound Load balancer Backend Address Pool for Secure Cluster Connectivity (No Public IP) workspace with managed virtual network. Changing this forces a new resource to be created.

- `sku` - (Required) The `sku` to use for the Databricks Workspace. Possible values are `standard`, `premium`, or `trial`.

Downgrading to a `trial sku` from a `standard` or `premium sku` will force a new resource to be created.

- `managed_services_cmk_key_vault_id` - (Optional) Resource ID of the Key Vault which contains the `managed_services_cmk_key_vault_key_id` key.

**Note:**

The `managed_services_cmk_key_vault_id` field is only required if the Key Vault exists in a different subscription than the Databricks Workspace. If the `managed_services_cmk_key_vault_id` field is not specified it is assumed that the `managed_services_cmk_key_vault_key_id` is hosted in the same subscriptioin as the Databricks Workspace.

**Note:**

If you are using multiple service principals to execute Terraform across subscriptions you will need to add an additional `azurerm_key_vault_access_policy` resource granting the service principal access to the key vault in that subscription.

- `managed_disk_cmk_key_vault_id` - (Optional) Resource ID of the Key Vault which contains the `managed_disk_cmk_key_vault_key_id` key

The `managed_disk_cmk_key_vault_id` field is only required if the Key Vault exists in a different subscription than the Databricks Workspace. If the `managed_disk_cmk_key_vault_id` field is not specified it is assumed that the `managed_disk_cmk_key_vault_key_id` is hosted in the same subscriptioin as the Databricks Workspace.

**Note:**

If you are using multiple service principals to execute Terraform across subscriptions you will need to add an additional `azurerm_key_vault_access_policy` resource granting the service principal access to the key vault in that subscription.

- `managed_services_cmk_key_vault_key_id` - (Optional) Customer managed encryption properties for the Databricks Workspace managed resources(e.g. Notebooks and Artifacts).

- `managed_disk_cmk_key_vault_key_id` - (Optional) Customer managed encryption properties for the Databricks Workspace managed disks.

- `managed_disk_cmk_rotation_to_latest_version_enabled` - (Optional) Whether customer managed keys for disk encryption will automatically be rotated to the latest version.

Databricks resources. Changing this forces a new resource to be created.

> ⚠ Note:
>
> Make sure that this field is unique if you have multiple Databrick Workspaces deployed in your subscription and choose to not have the `managed_resource_group_name` auto generated by the Azure Resource Provider. Having multiple Databrick Workspaces deployed in the same subscription with the same `manage_resource_group_name` may result in some resources that cannot be deleted.

- `customer_managed_key_enabled` - (Optional) Is the workspace enabled for customer managed key encryption? If `true` this enables the Managed Identity for the managed storage account. Possible values are `true` or `false`. Defaults to `false`. This field is only valid if the Databricks Workspace `sku` is set to `premium`.

- `infrastructure_encryption_enabled` - (Optional) Is the Databricks File System root file system enabled with a secondary layer of encryption with platform managed keys? Possible values are `true` or `false`. Defaults to `false`. This field is only valid if the Databricks Workspace `sku` is set to

- `public_network_access_enabled` - (Optional) Allow public access for accessing workspace. Set value to `false` to access workspace only via private link endpoint. Possible values include `true` or `false`. Defaults to `true`.

- `default_storage_firewall_enabled` - (Optional) Disallow public access to default storage account. Defaults to `false`.

- `access_connector_id` - (Optional) Access Connector ID to use when default storage account firewall is enabled.

> **Note:**
>
> The `access_connector_id` field is only required if `default_storage_firewall_enabled` is set to `true`.

- `network_security_group_rules_required` - (Optional) Does the data plane (clusters) to control plane communication happen over private link endpoint only or publicly? Possible values `AllRules`, `NoAzureDatabricksRules` or `NoAzureServiceRules`. Required when `public_network_access_enabled` is set to `false`.

- `tags` - (Optional) A mapping of tags to assign to the resource.

A `custom_parameters` block supports the following:

- `machine_learning_workspace_id` - (Optional) The ID of a Azure Machine Learning workspace to link with Databricks workspace. Changing this forces a new resource to be created.

- `nat_gateway_name` - (Optional) Name of the NAT gateway for Secure Cluster Connectivity (No Public IP) workspace subnets (only for workspace with managed virtual network). Defaults to `nat-gateway`. Changing this forces a new resource to be created.

- `public_ip_name` - (Optional) Name of the Public IP for No Public IP workspace with managed virtual network. Defaults to `nat-gw-public-ip`. Changing this forces a new resource to be created.

- `no_public_ip` - (Optional) Are public IP Addresses not allowed? Possible values are `true` or `false`. Defaults to `true`.

> ⚠ Note:
>
> Updating `no_public_ip` parameter is

> ⚠ Note:
>
> In `v3.104.0` and higher of the provider the `no_public_ip` parameter will now default to `true` instead of `false`.

- `public_subnet_name` - (Optional) The name of the Public Subnet within the Virtual Network. Required if `virtual_network_id` is set. Changing this forces a new resource to be created.

- `public_subnet_network_security_group_association_id` - (Optional) The resource ID of the `azurerm_subnet_network_security_group_association` resource which is referred to by the `public_subnet_name` field. This is the same as the ID of the subnet referred to by the `public_subnet_name` field. Required if `virtual_network_id` is set.

- `private_subnet_name` - (Optional) The name of the Private Subnet within the Virtual Network. Required if `virtual_network_id` is set. Changing this forces a new resource to be created.

- `private_subnet_network_security_group_association_id` - (Optional) The resource ID of the `azurerm_subnet_network_security_group_association` resource which is referred to by

- `storage_account_name` - (Optional) Default Databricks File Storage account name. Defaults to a randomized name(e.g. `dbstorage16mfeghoe5kxu`). Changing this forces a new resource to be created.

- `storage_account_sku_name` - (Optional) Storage account SKU name. Possible values include `Standard_LRS`, `Standard_GRS`, `Standard_RAGRS`, `Standard_GZRS`, `Standard_RAGZRS`, `Standard_ZRS`, `Premium_LRS` or `Premium_ZRS`. Defaults to `Standard_GRS`.

- `virtual_network_id` - (Optional) The ID of a Virtual Network where this Databricks Cluster should be created. Changing this forces a new resource to be created.

- `vnet_address_prefix` - (Optional) Address prefix for Managed virtual network. Defaults to `10.139`. Changing this forces a new resource to be created.

> ⚠ Note:
>
> Databricks requires that a network security group is associated with the `public` and `private` subnets when a `virtual_network_id` has been defined. Both `public` and `private` subnets must be delegated to `Microsoft.Databricks/workspaces`. For

## Example HCL Configurations

- Databricks Workspace Secure Connectivity Cluster with Load Balancer
- Databricks Workspace Secure Connectivity Cluster without Load Balancer
- Databricks Workspace with Private Endpoint
- Databricks Workspace with Private Endpoint, Customer Managed Keys for Managed Services and Databricks File System Customer Managed Keys
- Databricks Workspace with Root Databricks File System Customer Managed Keys
- Databricks Workspace with Root Databricks File System Customer Managed Keys in a Different Subscription
- Databricks Workspace with Customer Managed Keys for Managed Services
- Databricks Workspace with Customer Managed Keys for Managed Services with Key Vault and Key in a Different Subscription

## Attributes Reference

- `id` - The ID of the Databricks Workspace in the Azure management plane.
- `disk_encryption_set_id` - The ID of Managed Disk Encryption Set created by the Databricks Workspace.
- `managed_disk_identity` - A `managed_disk_identity` block as documented below.
- `managed_resource_group_id` - The ID of the Managed Resource Group created by the Databricks Workspace.
- `workspace_url` - The workspace URL which is of the format 'adb-{workspaceId}.{random}.azuredatabricks.net'
- `workspace_id` - The unique identifier of the databricks workspace in Databricks control plane.
- `storage_account_identity` - A `storage_account_identity` block as documented below.

A `managed_disk_identity` block exports the following:

- `principal_id` - The principal UUID for

workspace for enabling Customer Managed Keys.

- `tenant_id` - The UUID of the tenant where the internal databricks disks identity was created.
- `type` - The type of the internal databricks disks identity.

A `storage_account_identity` block exports the following:

- `principal_id` - The principal UUID for the internal databricks storage account needed to provide access to the workspace for enabling Customer Managed Keys.
- `tenant_id` - The UUID of the tenant where the internal databricks storage account was created.
- `type` - The type of the internal databricks storage account.

## Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 30 minutes) Used

- `update` - (Defaults to 30 minutes) Used when updating the Databricks Workspace.
- `read` - (Defaults to 5 minutes) Used when retrieving the Databricks Workspace.
- `delete` - (Defaults to 30 minutes) Used when deleting the Databricks Workspace.

## Import

Databrick Workspaces can be imported using the `resource id`, e.g.

```
terraform import azurerm_databr
```

Copy