

Getting started

HashiCorp Terraform is a popular opensource tool for creating safe and
predictable cloud infrastructure across
several cloud providers. You can use the
Microsoft Fabric Terraform Provider to
manage your Microsoft Fabric workspaces
using a flexible, powerful tool. The goal of
the Microsoft Fabric Terraform Provider is
to support automation of the most
complicated aspects of deploying and
managing Microsoft Fabric. Microsoft
Fabric customers are using the Microsoft
Fabric Terraform Provider to deploy and
manage clusters and jobs and to configure
data access.

In this section, you install and configure requirements to use Terraform and the Microsoft Fabric Terraform Provider on your local development machine. You then configure Terraform authentication.

Following this section, this article provides a sample configuration that you can

experiment with to provision a Microsoft Fabric Notebook and Lakehouse.

Requirements

- You must have the Terraform CLI. See Download Terraform on the Terraform website.
- 2. You must have a Terraform project. In your terminal, create an empty directory and then switch to it. (Each separate set of Terraform configuration files must be in its own directory, which is called a Terraform project.) For example:

Include Terraform configurations for your project in one or more configuration files in your Terraform project. For information about the configuration file syntax, see Terraform Language Documentation on the Terraform website.

- You must configure authentication for your Terraform project. See Authentication in the Microsoft Fabric Terraform Provider documentation.
- 4. You must have a Fabric Capacity

provisioned in Azure. See Configuring a Fabric Capacity in the Microsoft Fabric Terraform Provider documentation.



Note

Please keep the capacity name handy, as we will use it below to fetch the capacity id.

Sample configuration

This section provides a sample configuration that you can experiment with to provision a Microsoft Fabric Notebook and a Lakehouse. It assumes that you have already set up the requirements, as well as created a Terraform project and configured the project with Terraform authentication as described in the previous section.

- 1. Create a new file named provider.tf in your Terraform project directory.
- 2. Add the following code to provider.tf to define a dependency on the Microsoft Fabric Terraform Provider:

```
# We strongly recommen Copy
terraform {
    required_version = ">= 1.8, <
    required_providers {
        fabric = {
            source = "microsoft/fabr
            version = "1.3.0"
        }
    }
}
# Configure the Microsoft Fabri
provider "fabric" {
    # Configuration options
}</pre>
```

3. Create another file named

variables.tf, and add the following code. This file represents input variables that can be used to configure a Notebook and Lakehouse.

```
variable "workspace_di Copy am
  description = "A name for the
  type = string
}
variable "notebook_display_name
  description = "A name for the
  type
        = string
}
variable "notebook_definition_u
  description = "Whether to upd
            = bool
  type
  default
            = true
}
```

FABRIC DOCUMENTATION

Q Filter

Service
Principal and
Client
Certificate

Authenticating using a Service Principal and Client Secret

Authenticating using a Service Principal and OpenID Connect (OIDC)

Creating an
App
Registration
for the Service
Principal
context (SPN)

Creating an App Registration for the User context

Guides

```
variable "notebook_definition_p
  description = "The path to th
  type = string
}

variable "capacity_name" {
  description = "The name of th
  type = string
}
```

4. Create a file named workspace.tf and add the following hcl code to represent a Microsoft Fabric workspace. We will also add a data source to fetch the Microsoft Fabric Capacity id by name (see requirements section).

```
data "fabric_capacity" Copy it
  display_name = var.capacity_n
}

resource "fabric_workspace" "ex
  display_name = var.workspace_
  description = "Getting starte
  capacity_id = data.fabric_cap
}
```

- 5. Create a file named notebook.ipynb in the same folder and copy the content of this example notebook.
- 6. Create a file named notebook.tf and add the following hcl code to represent a Notebook. This Notebook references the workspace created in

```
Requirements
Sample
configuration
```

Troubleshooting

Report an issue 🗷

Advanced use cases with Go templating

Configuring a

step 4, specifically using the workspace id.

```
resource "fabric_noteb Copy ka
  workspace_id = fabric_workspa
  display_name = var.notebook_d
  definition_update_enabled = v
  definition = {
    "notebook-content.ipynb" =
        source = var.notebook_def
    }
}
```

7. Create another file named

terraform.tfvars , and add the following code. This file specifies the Notebook's properties. Learn more about tfvars file.

8. Create another file named

outputs.tf , this is where we will define Terraform output values. Add the following code:

```
output "capacity_id" {       Copy
    value = data.fabric_capacity.
}
output "notebook_id" {
```

```
value = fabric_notebook.examp
}
```

- 9. Run terraform init. If there are any errors, fix them, and then run the command again.
- 10. Run terraform plan out=plan.tfplan . If there are any
 errors, fix them, and then run the
 command again. In this example, we
 are capturing the output to a plan file
 named plan.tfplan .
- 11. Run terraform apply plan.tfplan.

 This command applies the changes required to reach the desired state of the configuration. If there are any errors, fix them, and then run the command again.
- 12. Verify that the Workspace and
 Notebook were created in Microsoft
 Fabric. In the output of the terraform
 apply command, find the Notebook
 id and capacity id.
- when you are done with this sample, delete the Notebook, and workspace from Microsoft Fabric by running

```
terraform destroy .
```

Troubleshooting

See Troubleshooting guide.

INTRO



© HashiCorp 2025

LEARN

DOCS

EXTEND

COMMUNITY

STATUS

PRIVACY

SECURITY

TERMS

PRESS KIT