Registry

🔍 Search all resources

Providers / microsoft / fabric / Version 1.3.0 ⌄   Latest Version

fabric 🤝   Overview   Documentation   USE PROVIDER ▼

Microsoft Fabric Provider

The Microsoft Fabric Provider allows managing environments and other resources within Microsoft Fabric.

Requirements

This provider requires **Terraform >= 1.8.x** (>= 1.11.x preferred). For more information on provider installation and constraining provider versions, see the Provider Requirements documentation.

Installation

To install this provider, copy and paste this code into your Terraform configuration. Then, run `terraform init`.

Report an issue 🗗

```
# We strongly recommend us...                Copy
terraform {
```

```
  required_version = ">= 1.8, < 2.0
  required_providers {
    fabric = {
      source  = "microsoft/fabric"
      version = "1.3.0"
    }
  }
}


# Configure the Microsoft Fabric Te
provider "fabric" {
  # Configuration options
}
```

Installation (developers only)

To use the provider you can download the binaries from Releases to your local file system and configure Terraform to use your local mirror. See the Explicit Installation Method Configuration for more information about using local binaries.

```
# Explicit Installation Me        nf                    Copy
# docs: https://developer.hashicorp

provider_installation {
  filesystem_mirror {
    path    = "/usr/share/terraform
    include = ["registry.terraform.
  }
}
```

Authentication

The provider allows authentication via service principal or user credentials. All sensitive information should be passed into Terraform using environment variables (don't put secrets in your tf files).

Using Azure CLI (Default)

The Fabric provider can use the Azure CLI to authenticate. If you have the Azure CLI installed, you can use it to log in to your Azure account and the Fabric provider will use the credentials from the Azure CLI.

1. Install the Azure CLI

2. Follow the Creating an App Registration for the User context to use with Azure CLI guide.

Using a Service Principal

You can find more information on how to do this in the following guides:

- Authenticating using Managed Identity (MSI)

- Authenticating using a Service Principal and OpenID Connect (OIDC)

- Authenticating using a Service Principal and Client Certificate

- [Authenticating using a Service Principal and Client Secret](#)

Schema

Optional

- `auxiliary_tenant_ids` (Set of String) The Auxiliary Tenant IDs which should be used.

- `azure_devops_service_connection_id` (String) The Azure DevOps Service Connection ID that uses Workload Identity Federation.

- `client_certificate` (String, Sensitive) Base64 encoded PKCS#12 certificate bundle. For use when authenticating as a Service Principal using a Client Certificate.

- `client_certificate_file_path` (String) The path to the Client Certificate associated with the Service Principal for use when authenticating as a Service Principal using a Client Certificate.

- `client_certificate_password` (String, Sensitive) The password associated with the Client Certificate. For use when authenticating as a Service Principal using a Client Certificate.

- `client_id` (String) The Client ID of the app registration.

- `client_id_file_path` (String) The path to a file containing the Client ID which should be used.

- `client_secret` (String, Sensitive) The Client Secret of the app registration. For use when authenticating as a Service Principal using a Client Secret.

- `client_secret_file_path` (String) The path to a file containing the Client Secret which should be used. For use when authenticating as a Service Principal using a Client Secret.

- `disable_terraform_partner_id` (Boolean) Disable sending the Terraform Partner ID if a custom `partner_id` isn't specified, which allows Microsoft to better understand the usage of Terraform. The Partner ID does not give HashiCorp any direct access to usage information. This can also be sourced from the `FABRIC_DISABLE_TERRAFORM_PARTNER_ID` environment variable. Defaults to `false`.

- `endpoint` (String) The Endpoint of the Microsoft Fabric API.

- `environment` (String) The cloud environment which should be used.

Possible values are 'public', 'usgovernment' and 'china'. Defaults to 'public'

- `oidc_request_token` (String, Sensitive) The bearer token for the request to the OIDC provider. For use when authenticating as a Service Principal using OpenID Connect.

- `oidc_request_url` (String) The URL for the OIDC provider from which to request an ID token. For use when authenticating as a Service Principal using OpenID Connect.

- `oidc_token` (String, Sensitive) The OIDC token for use when authenticating as a Service Principal using OpenID Connect.

- `oidc_token_file_path` (String) The path to a file containing an OIDC token for use when authenticating as a Service Principal using OpenID Connect.

- `partner_id` (String) A GUID/UUID that is registered with Microsoft to facilitate partner resource usage attribution.

- `preview` (Boolean) Enable preview mode to use preview features.

- `tenant_id` (String) The ID of the Microsoft Entra ID tenant that Fabric

API uses to authenticate with.

- `timeout` (String) Default timeout for all requests. It can be overridden at any Resource/Data-Source A string that can be [parsed as a duration](#) consisting of numbers and unit suffixes, such as `30s` or `2h45m`. Valid time units are "s" (seconds), "m" (minutes), "h" (hours) If not set, the default timeout is `10m`.

- `use_cli` (Boolean) Allow Azure CLI to be used for authentication.

- `use_dev_cli` (Boolean) Allow Azure Developer CLI to be used for authentication.

- `use_msi` (Boolean) Allow Managed Service Identity (MSI) to be used for authentication.

- `use_oidc` (Boolean) Allow OpenID Connect to be used for authentication.

Known limitations

- **Capacity**: [Microsoft Fabric trial capacity](#) is not supported. Only self-provisioned [Fabric Capacity](#) on Azure is supported. You can setup your capacity in the [Azure Portal](#).

- **Service Principal**: Not all Fabric

resources support Service Principals yet. For Provider evaluation, we recommend using the Azure CLI for authentication with User context.

Contributing

This project welcomes feedback and suggestions only via GitHub Issues. Pull Request (PR) contributions will **NOT** be accepted at this time. Please see the Contribution Guidelines

INTRO

LEARN

DOCS

EXTEND

COMMUNITY

STATUS

PRIVACY

SECURITY

TERMS

PRESS KIT

© HashiCorp 2025