Using Azure for Remote State

Ned Bellavance

MICROSOFT AZURE MVP

@ned1313 | nedinthecloud.com

Overview

Terraform remote state

Azure Storage for remote state

Migrating the current state
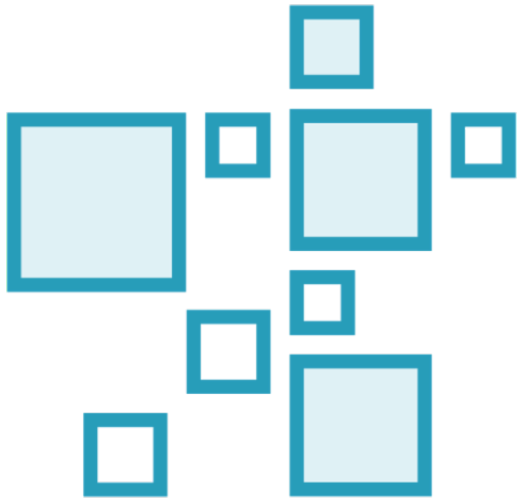
Terraform Remote State

Safeguard the state file

Enable team collaboration

Multiple supported backends

Special features
- Locking
- Workspaces

Azure Storage for Remote State

Azure Blob Storage

Supports locking

Supports workspaces

Multiple auth methods

Authentication Methods

Managed Service Identity

use_msi

Shared access signature token

sas_token
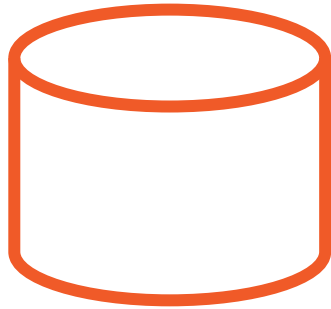
Storage access key

access_key

Service principal
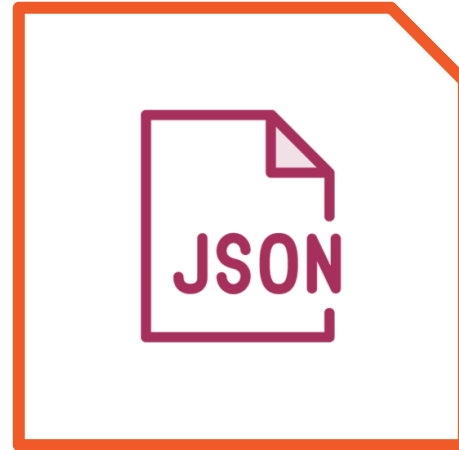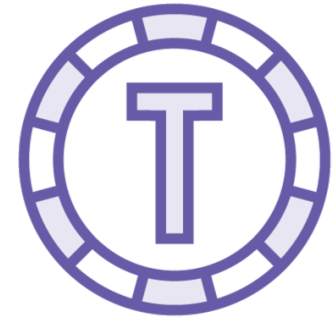
client_id

Remote State Storage

Storage account

Container

SAS Token

Migrating Terraform State

Update backend configuration

Run terraform init

Confirm state migration

Backend Configuration

```
terraform {

  backend "azurerm" {

    storage_account_name = "statestore123098"

    container_name       = "terraform-state"

    key                  = "terraform.tfstate"

  }

}

terraform init -backend-config="sas_token=sdfghrtyu265"
```

Summary

Use remote state by default

Azure Storage supports locking and workspaces

Migration is super simple

Coming up:

- Adding in automation

- Adopting Azure DevOps