

Terraform State Introduction

Terraform State

- Terraform state is the memory of Terraform
 - Acts as an internal storage - holds your infrastructure state, metadata, resource attributes, internal dependencies, etc.
- Think of it like the human brain - holding memories, connecting nerves and receiving information from systems throughout the body



Terraform State

- Terraform State serves three main purposes: Mapping, Metadata, Performance
- Mapping
 - This is straight forward - acts like an internal database to map defined configuration with real world data
 - Example: EC2 instance id to your EC2 instance resource
 - Example: Defined VPC cidr blocks to your VPC resource
- Metadata
 - Helps maintain internal dependency ordering
 - If resource A depends on resource B, and resource B is removed. What happens?
 - Metadata for resource creation - which provider was used? which alias?
- Performance
 - Acts as a caching mechanism of resource attributes
 - Easy to read values internally rather than fetching values over the network

Terraform State

- Terraform State helps Terraform make smart decisions on infrastructure changes
 - State allows Terraform to easily compare changes between config vs. real world
 - Helps keep track of resource attributes and metadata
 - Improves Terraform performance, by acting as an internal cache
- There are two ways to host state files
 - **Local**: The default setting, stores state locally at the calling config directory
 - **Remote**: Offloading state to a centralized storage (remote backend) like S3
- Terraform automatically creates and manages the statefile for you
 - However, there are cases where you may need to inspect or edit the state
 - Terraform provides state commands for reading and editing the state

Terraform State

- Terraform stores state locally in a file called `terraform.tfstate` and auto maintains a backup
 - But this poses a problem
 - If more than one person is working on Terraform, how is the statefile managed?
 - Is it safe to store the statefile in a repository along with your code?
- Statefiles may contain sensitive metadata and attributes regarding your infrastructure
 - Not safe to store this in your repository - plaintext file
 - Any sensitive data should never be stored in a repository
- Local statefiles make it extremely difficult to work on Terraform in a team
 - Impossible to manage and not scalable
- The best practice is to always use remote states whenever possible