

Terraform Init Advanced

Terraform Init Advanced

- Default behavior recap:
 - Installs and caches defined Providers and Modules
 - Pulls in any recent changes relating to Providers and Modules
 - Initializes the backend which contains the state files
 - Creates a provider version lock file
- Need to know three additional flags:
 - **terraform init -reconfigure**
 - Terraform initializes the backend which hosts your state files
 - If the backend configuration changes or a new backend is used
 - **terraform init -upgrade**
 - Terraform records and locks version in a dependency lock file
 - If Provider version needs to be changed, we need to update the dependency lock file
 - **terraform init -migrate-state**
 - When transitioning from a local state to remote, or a remote state to new backend
- In 90% of cases, **terraform init** is enough and no additional flags are needed

Terraform Init Advanced

- **terraform init** creates a local folder **.terraform** to store Provider and Module binaries
 - Acts as a local cache to improve performance and stability
- If facing strange issues with running **terraform init** - delete the **.terraform** folder
 - Removes the cache
 - Caching can sometimes cause unintended issues, difficult to debug
- Removing the **.terraform** folder provides clean slate and a new local cache