Terraform

About

- Welcome to Terraform course.
- We would move from very basic level to the advance level.
- This course is for the absolute beginners, for people with little experience or more working experience who want to enhance their skills.
- Complete hands-on work with less theory.
- This course would have more than 7hrs of quality content
- Real time projects experience
- Best live examples with multi-Cloud examples i.e., AWS, Azure, GCP

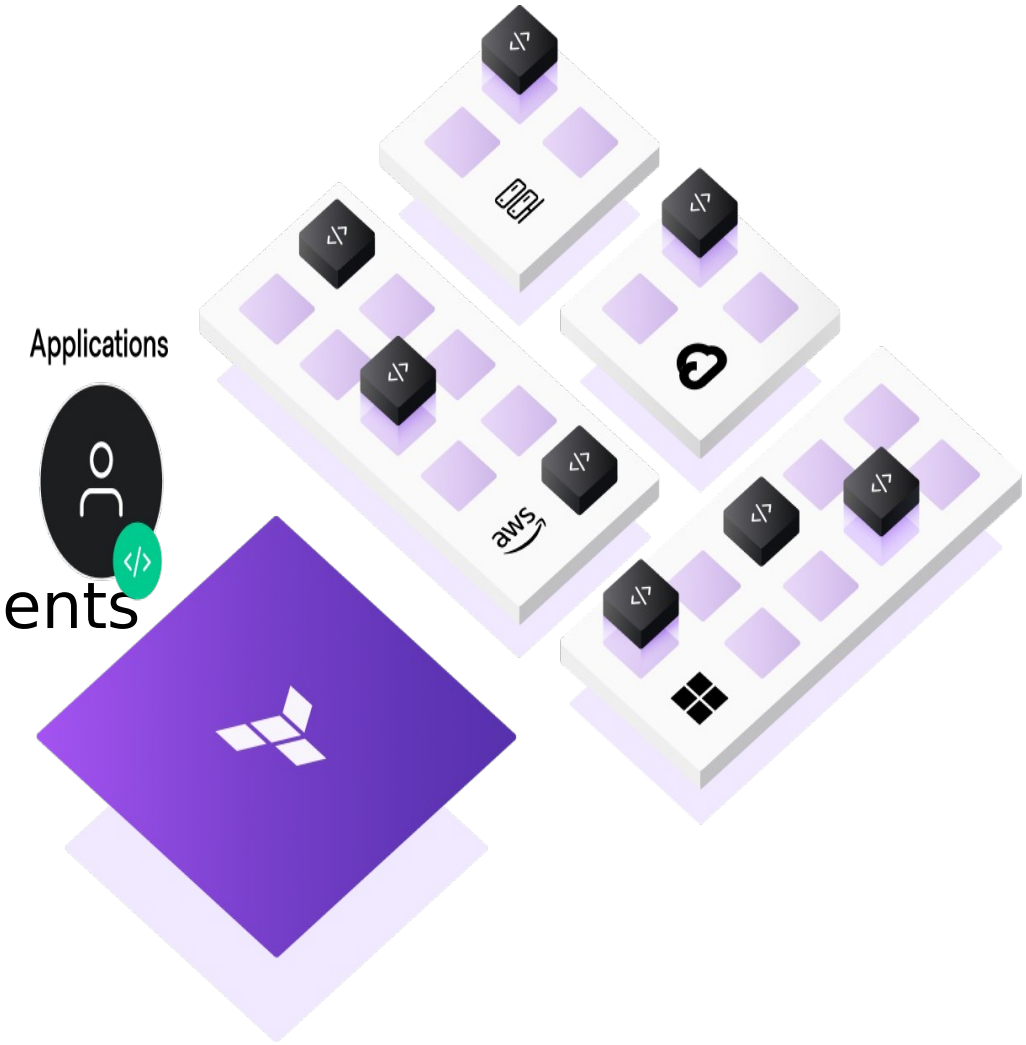
Terraform: by al-nafi

What is IAC

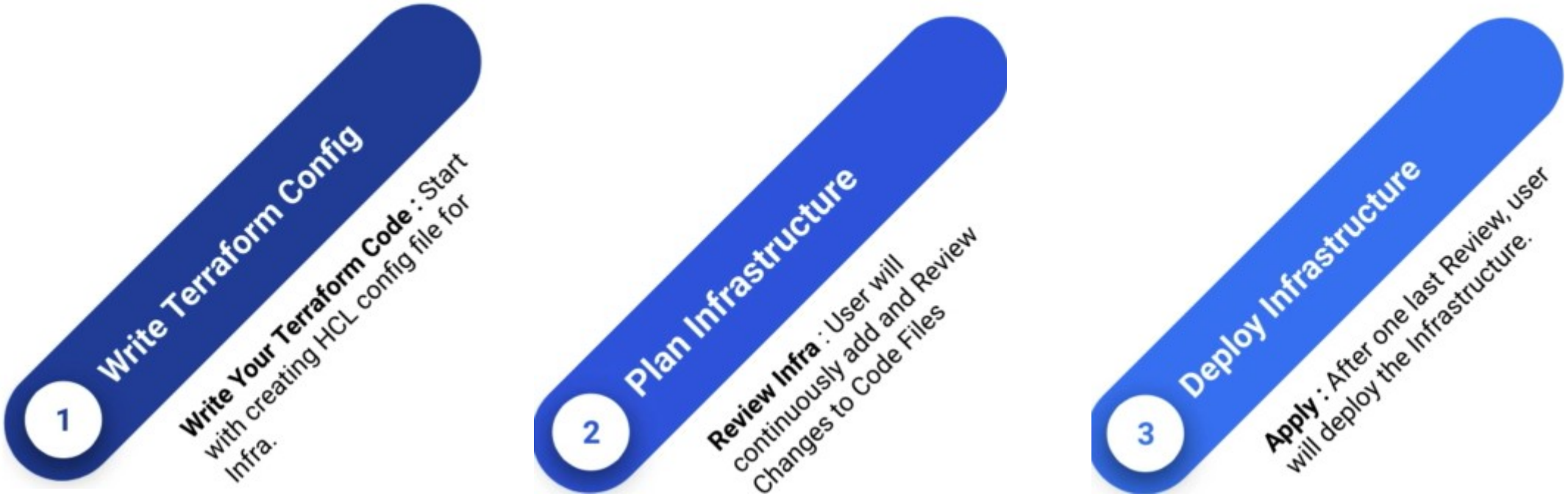
- IAC: Infrastructure as a code
- No manual Deployment: Write anything you want to deploy(vms ,disks, app etc.) Human readable code
- DevOps Enablement: Deployment codification means it can be versioned through VCS
- Create your own infrastructure
- Speed, cost and risk: less human intervention during deployment; fewer chances of error, miss configuration, redeployment and more time is saved

IAC with Terraform

- Automate Software defining Deployment Network
- Supports vast range of public/private Cloud Providers
- Use terraform to deploy Environments across the multiple clouds
- Track state of each resource Deployed

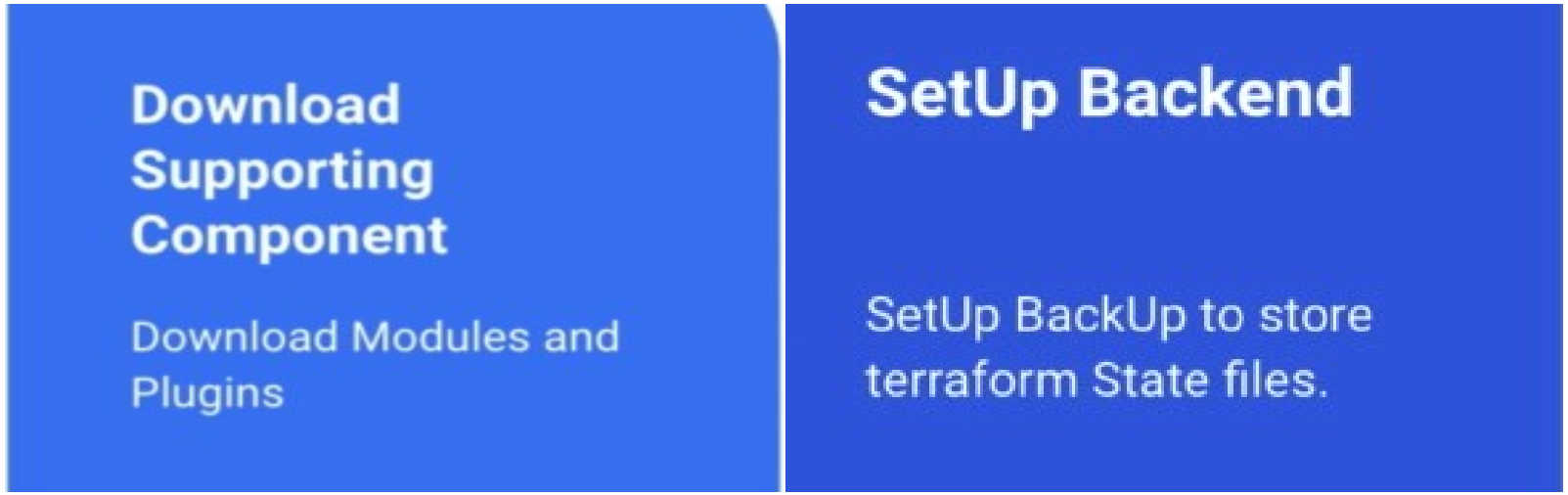


Terraform Workflow



Terraform Initialization

- Terraform init: command to initialize the working directory that contains your code.



Terraform Key Concepts

- Terraform plan:

- After initialization, it reads code, creates and shows a plan for execution/deployment

- It doesn’t deploy anything; it shows what is going to deploy.
- It allows user to review the Action plan before executing anything
- Authentication credentials are used to connect to your infrastructure.

Terraform Key Concepts

- Terraform deploy:

- Deploy the infrastructure and statements in the code
- Update the deployment state tracking i.e., state files.
- If some resources are already deployed. This command will deploy updated code and tracking.

Terraform Key Concepts

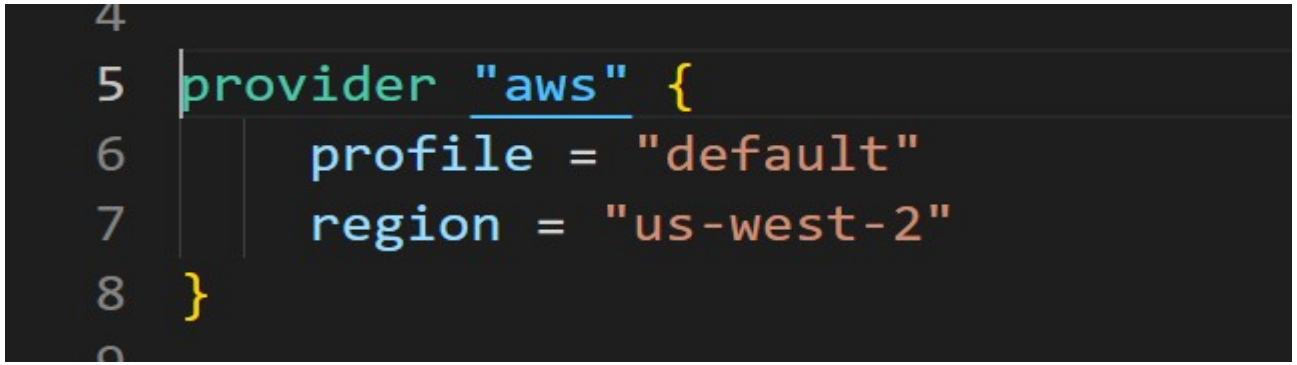
- Terraform destroy:

- Looks at the recorded and store state files created during deployment and destroy all resources found in state file.
- Should be used very carefully because it is non-reversible command. Take backup and make sure what you want to destroy.

Write IAC | **Terraform Init** | **Terraform Plan** | **Terraform Apply** | **Terraform Destroy**

Terraform Providers

- Providers are public cloud Vendors, to which Terraform interact to create resources.
- Terraform relies on plugins called “providers” to interact with cloud providers.
- Terraform configurations must declare which providers they requires so that Terraform can install and use them.



Terraform Providers

- Terraform finds and install providers as you Initialize Terraform(using terraform init command)

- As best Practice Providers should be versioned in your Terraform config files.

- Terraform Providers release is separate from Terraform release

Terraform: AWS Setup

- Spinning Instance on AWS.
- Need AWS account.
- Create IAM admin user.
- Manage security in Terraform.

Terraform State

- Terraform state file are very important for resource tracking
- Terraform state a way that terraform read to identify what has been deployed
- Very critical to terraform functionality
- Stored in flat file by default name “terraform.tfstate”
- Stored in same working directory but can also be stored remotely
- Helps terraform calculate Terraform Deltas. Terraform plan can be used to create the new Development Plan
- WARNING: NEVER LOSE YOUR TERRAFORM STATE FILE