Terraform: Infrastructure as Code

Providers

Terraform Providers

There are 200+ available providers for a broad set of common infrastructure. Provider SDK makes it simple to create new and custom providers.

Providers Architecture

- Provider code is a very subtle layer for cloud or service API.
- Providers themselves are executable files that communicate with TF via gRPC.
- Each Resource implements CREATE, READ, UPDATE, and DELETE (CRUD) methods to manage itself, while Terraform Core manages a Resource Graph of all the resources declared in the configuration as well as their current state.

Provider Configuration

A provider configuration is created using a provider block:

```
provider "aws" {
  version = "~> 2.0"
  region  = "us-east-1"
}
```

The name given in the block header ("aws" in this example) is the name of the provider to configure.

The body of the block (between { and }) contains configuration arguments for the provider itself.

terraform init

```
...
Initializing provider plugins...
- Checking for available provider plugins...
- Downloading plugin for provider "aws" (hashicorp/aws) 2.59.0...

The following providers do not have any version constraints in configuration,
so the latest version was installed.
...
```

Modules: Registry

Registry source address: `<NAMESPACE>/<NAME>/<PROVIDER>`

```
module "consul" {
  source = "hashicorp/consul/aws"
  version = "0.1.0"
}
```

Registry: Requirements

- GitHub. The module must be on GitHub and must be a public repo
- Named `terraform-<PROVIDER>-<NAME>`
- Repository description
- Standard module structure. The module must adhere to the standard module structure
- x.y.z tags for releases