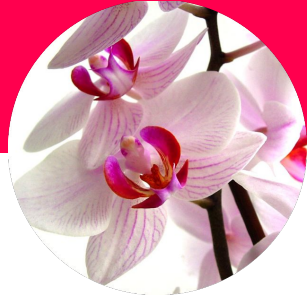


The background of the slide features a close-up of cherry blossoms. The upper portion is covered by a semi-transparent red filter, through which the white and pink petals and dark branches are visible. The lower portion of the slide shows the blossoms in their natural colors, with white petals and prominent pink stamens.

# **| How .tf to do | Infrastructure as Code**



# Hello!

Alix Klingenberg  
Duck Lawn





# Hello!

Alix Klingenberg  
Duck Lawn



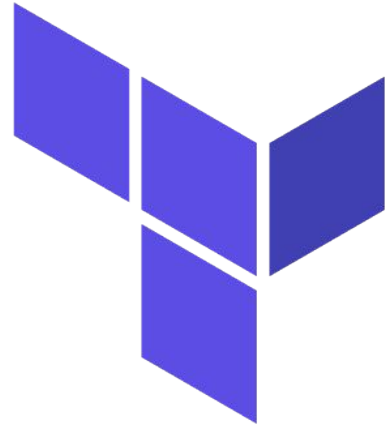
# Infrastructure as Code

Reusable

Repeatable

Auditable

Shared



HashiCorp

**Terraform**

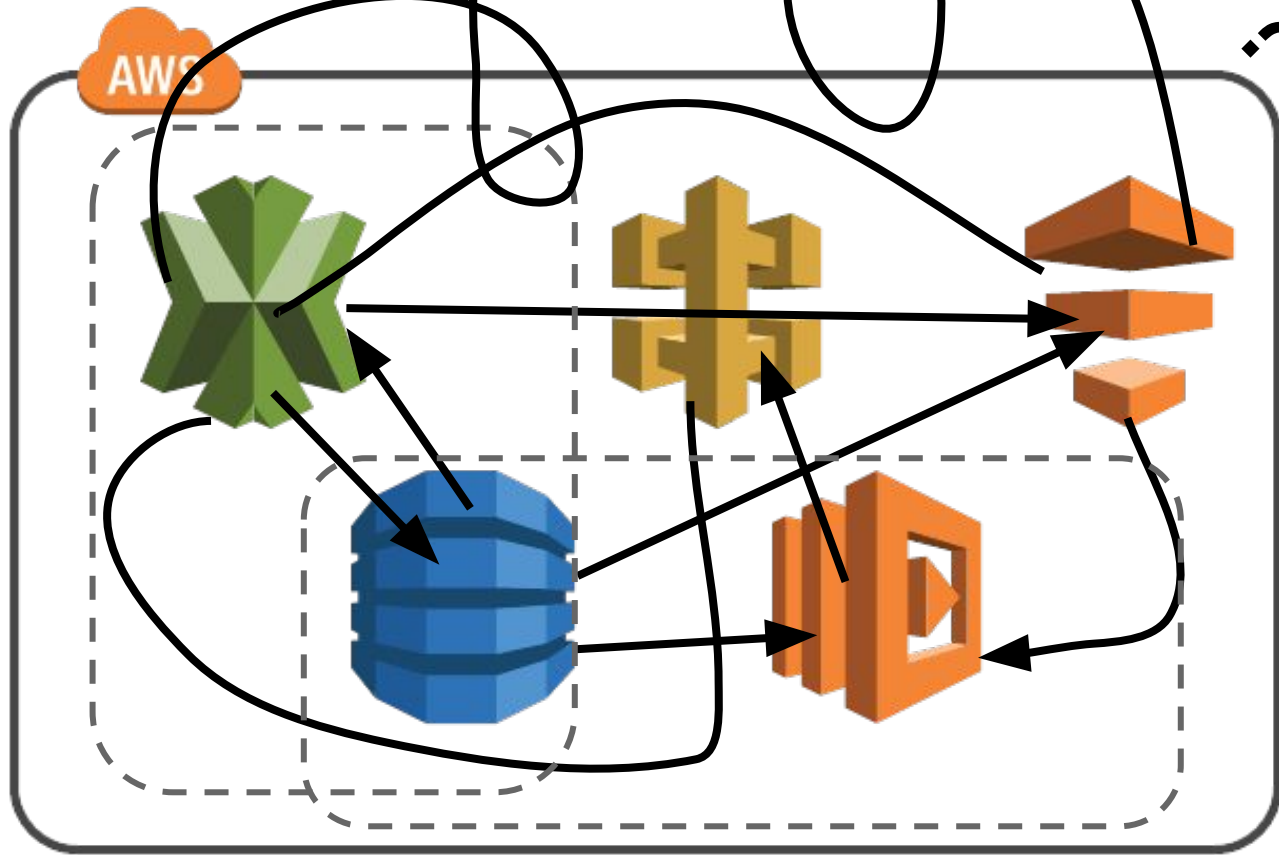
“

*How tho?*



**All the things!**

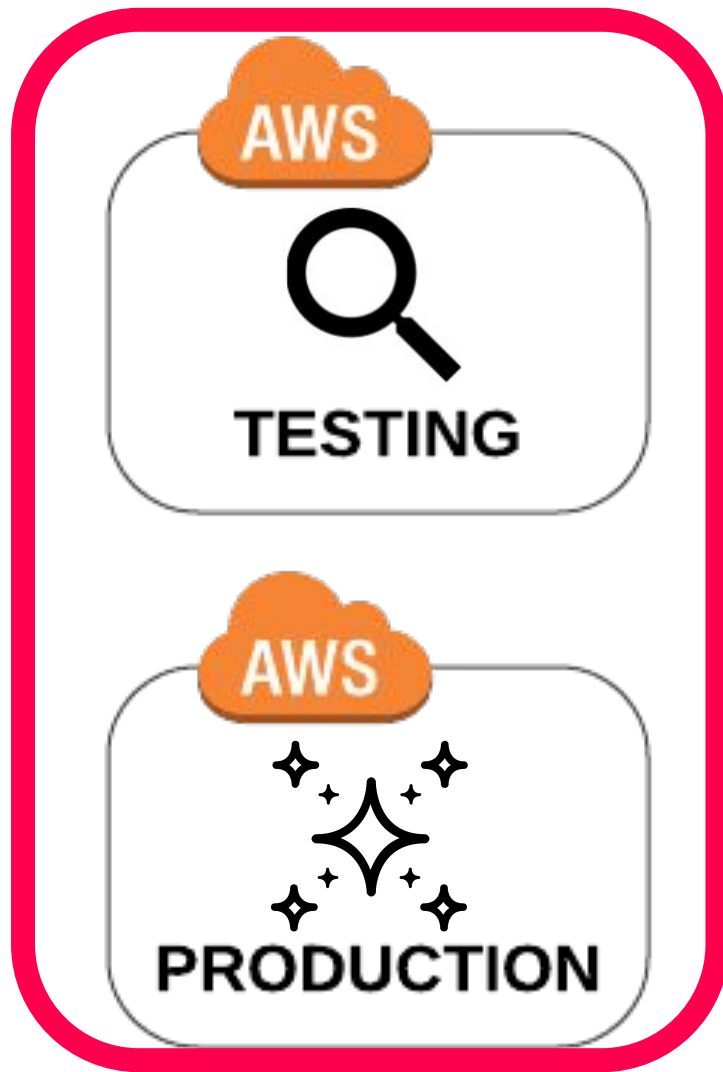
$$\Omega \mu + \mathcal{M} = a/c$$



$$1+2=\$$$









# “ Identity and Access Management (IAM)



aka all your users and stuff

# IAM policy



```
statement {  
    effect = "Allow",  
    actions = [  
        "s3:PutObject",  
        "s3:GetObject",  
    ],  
    resources = "arn:aws:s3:::my-bucket/*",  
}
```



**IAM Role**

# IAM Role - trust policy

```
statement {  
    effect = "Allow",  
    actions = "sts:AssumeRole",  
    principals {  
        type = "AWS",  
        identifier = ["lambda.amazonaws.com"],  
    }  
}
```



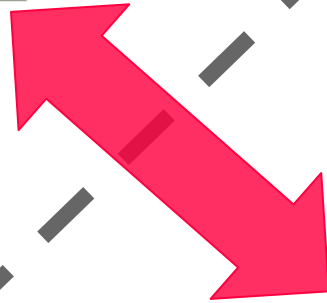
Admin

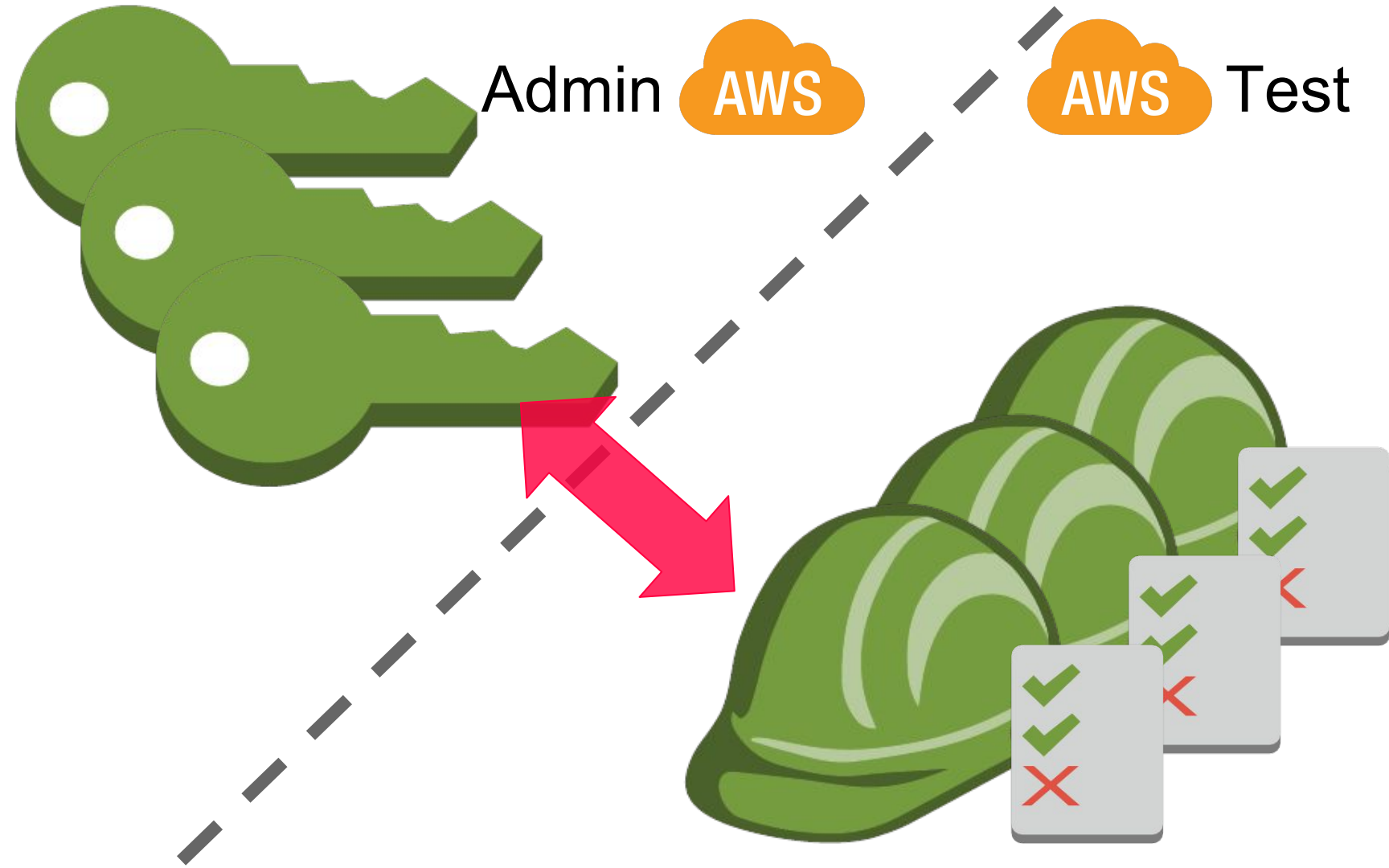
AWS

AWS

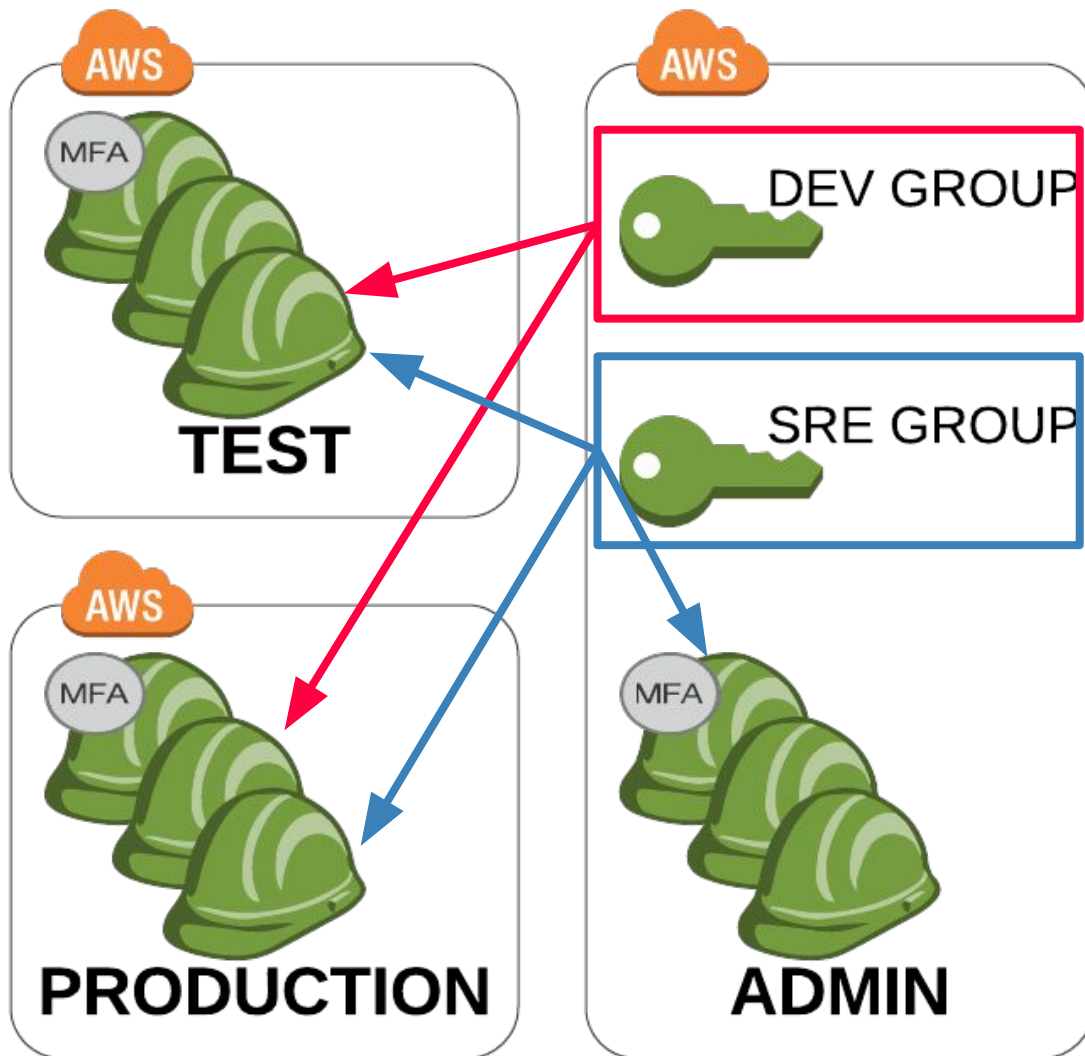
Test

IAM Group









AWS

MFA



**TEST**

AWS



DEV GROUP



ALL USERS  
GROUP



SRE GROUP

AWS

MFA



**PRODUCTION**



MFA



**ADMIN**

# LIVE DEMO



Account ID or alias

IAM user name

Password

[Sign-in using root account credentials](#)



The banner features a dark blue background with a stylized illustration of a cloud at the top center. On either side of the cloud are database cylinder icons with three vertical lines below them, representing data flow. In the center, below the cloud, is a large white bracket containing the numbers '90000' in blue. Below this illustration, the text 'AWS Database Migration Service' is written in large, bold, white font. Underneath that, 'Over 90,000 Databases Migrated' is written in a smaller, blue, italicized font. Below that, 'Easily migrate and convert databases' is written in white. At the bottom center, there is a white rectangular button with the text 'Learn more »' in blue.

# AWS Database Migration Service

*Over 90,000 Databases Migrated*

Easily migrate and convert databases

Learn more »

You said there would be **code...**



```
1 terraform {
2     required_version = ">= 0.11.7"
3
4     backend "s3" {
5         encrypt = "true"
6         bucket  = "my-first-terraform-test"
7         key     = "test/terraform.tfstate"
8         region  = "us-west-2"
9     }
10 }
11
12 provider "aws" {
13     version           = "1.40.0"
14     region            = "us-west-2"
15     allowed_account_ids = ["123456789"]
16 }
```

```
1 terraform {
2     required_version = ">= 0.11.7"
3
4     backend "s3" {
5         encrypt = "true"
6         bucket  = "my-first-terraform-test"
7         key     = "test/terraform.tfstate"
8         region  = "us-west-2"
9     }
10 }
11
12 provider "aws" {
13     version           = "1.40.0"
14     region            = "us-west-2"
15     allowed_account_ids = ["123456789"]
16 }
```



```
1 terraform {
2     required_version = ">= 0.11.7"
3
4     backend "s3" {
5         encrypt = "true"
6         bucket  = "my-first-terraform-test"
7         key     = "test/terraform.tfstate"
8         region  = "us-west-2"
9     }
10 }
11
12 provider "aws" {
13     version           = "1.40.0"
14     region            = "us-west-2"
15     allowed_account_ids = ["123456789"]
16 }
```





# What is a **tf state file**

Amazon S3 > my-first-terraform-test / test

terraform.tfstate Latest version

Overview

Properties

Open

Download

Download

**Owner**

duckalini

**Last modified**

Aug 4, 2018 6:40:17 PM GMT+1200

**Etag**

8ed0be6141d04f4b4f5c491c3795cbe8

**Storage class**

Standard

**Server-side encryption**

AES-256

**Size**

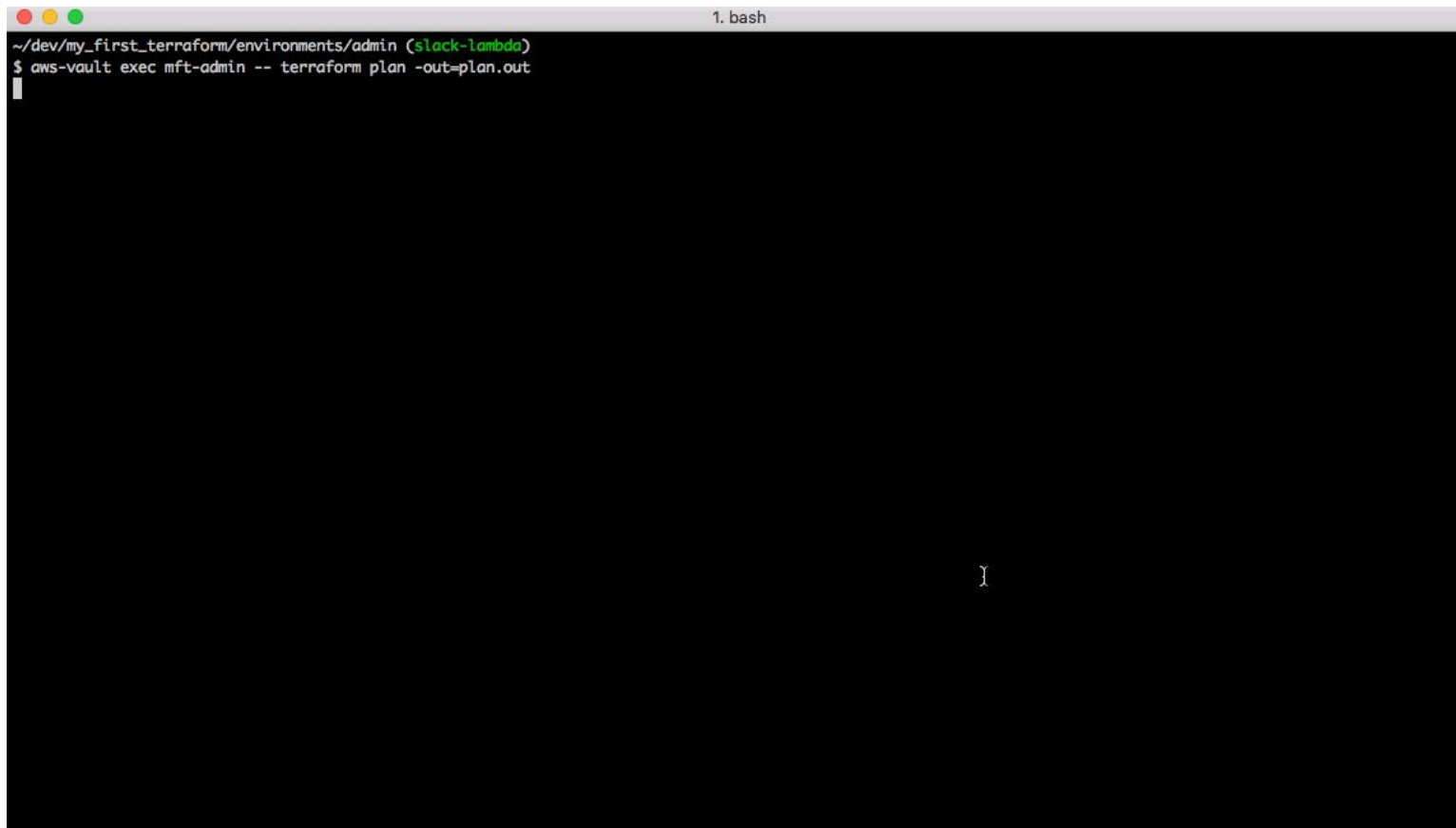
317

```
$ cat terraform.json
{
  "version": 3,
  "terraform_version": "0.11.7",
  "serial": 1,
  "lineage": "a1d176ea-1465-e6c1-05cf-7d90a114bf5e",
  "modules": [
    {
      "path": [
        "root"
      ],
      "outputs": {},
      "resources": {},
      "depends_on": []
    }
  ]
}
```

| **.tf files + .tfstate file**  
[local] [remote]

**= plan diff**


# Now we terraform plan -out=plan.out

A terminal window with a light gray title bar containing three colored window control buttons (red, yellow, green) on the left and the text "1. bash" on the right. The terminal background is black with white text. The prompt is "~/dev/my\_first\_terraform/environments/admin (slack-lambda)". The command entered is "\$ aws-vault exec mft-admin -- terraform plan -out=plan.out". A white cursor is positioned at the end of the command line.

```
~/dev/my_first_terraform/environments/admin (slack-lambda)
$ aws-vault exec mft-admin -- terraform plan -out=plan.out
```


```
+ module.slack.aws_sns_topic_subscription.sns_notify_slack
  id:                                     <computed>
  arn:                                     <computed>
  confirmation_timeout_in_minutes:        "1"
  endpoint:                               "${aws_lambda_function.notify_slack.0.
  endpoint_auto_confirms:                  "false"
  protocol:                                "lambda"
  raw_message_delivery:                    "false"
  topic_arn:                              "${aws_sns_topic.slack_notification.arn
```

Plan: 6 to add, 0 to change, 0 to destroy.



-----

This plan was saved to: plan.out




To perform exactly these actions, run the following command to apply:

```
terraform apply "plan.out"
```

## Now we terraform apply plan.out

```
principal:      "" => "sns.amazonaws.com"
source_arn:     "" => "arn:aws:sns:us-west-2:867697617212:slack_notification"
statement_id:   "" => "AllowExecutionFromSNS"
module.slack.aws_sns_topic_subscription.sns_notify_slack: Creating...
  arn:          "" => "<computed>"
  confirmation_timeout_in_minutes: "" => "1"
  endpoint:     "" => "arn:aws:lambda:us-west-2:867697617212"
  endpoint_auto_confirms: "" => "false"
  protocol:     "" => "lambda"
  raw_message_delivery: "" => "false"
  topic_arn:    "" => "arn:aws:sns:us-west-2:867697617212:slack_notification"
module.slack.aws_lambda_permission.sns_notify_slack: Creation complete after 2s
module.slack.aws_sns_topic_subscription.sns_notify_slack: Creation complete after 2s
lac...n:7063feee-8255-4d1f-aa72-e2c6220acae3)

Apply complete! Resources: 6 added, 0 changed, 0 destroyed.
```

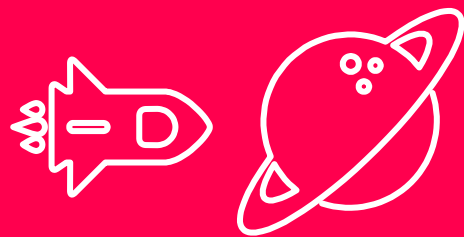


Now you can start **building!**



**Modules and  
code and layouts  
oh my!**





# The benefits

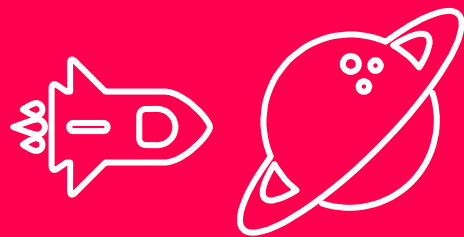
Reusability

Repeatability

Auditability

Shared responsibility



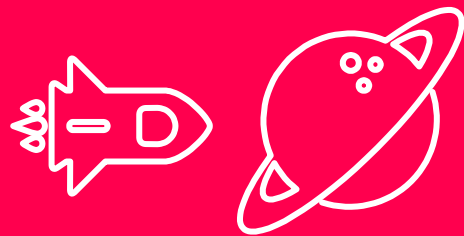


# How it works

Statefiles

Planning and applying changes

How to create modules



# Reusable code!


You can find all terraform code used to build these environments at  
[https://github.com/duckalini/my\\_first\\_terraform](https://github.com/duckalini/my_first_terraform)



# Thanks!

Alix Klingenberg

Duck Lawn

 @evolutionises

 @duckalini



