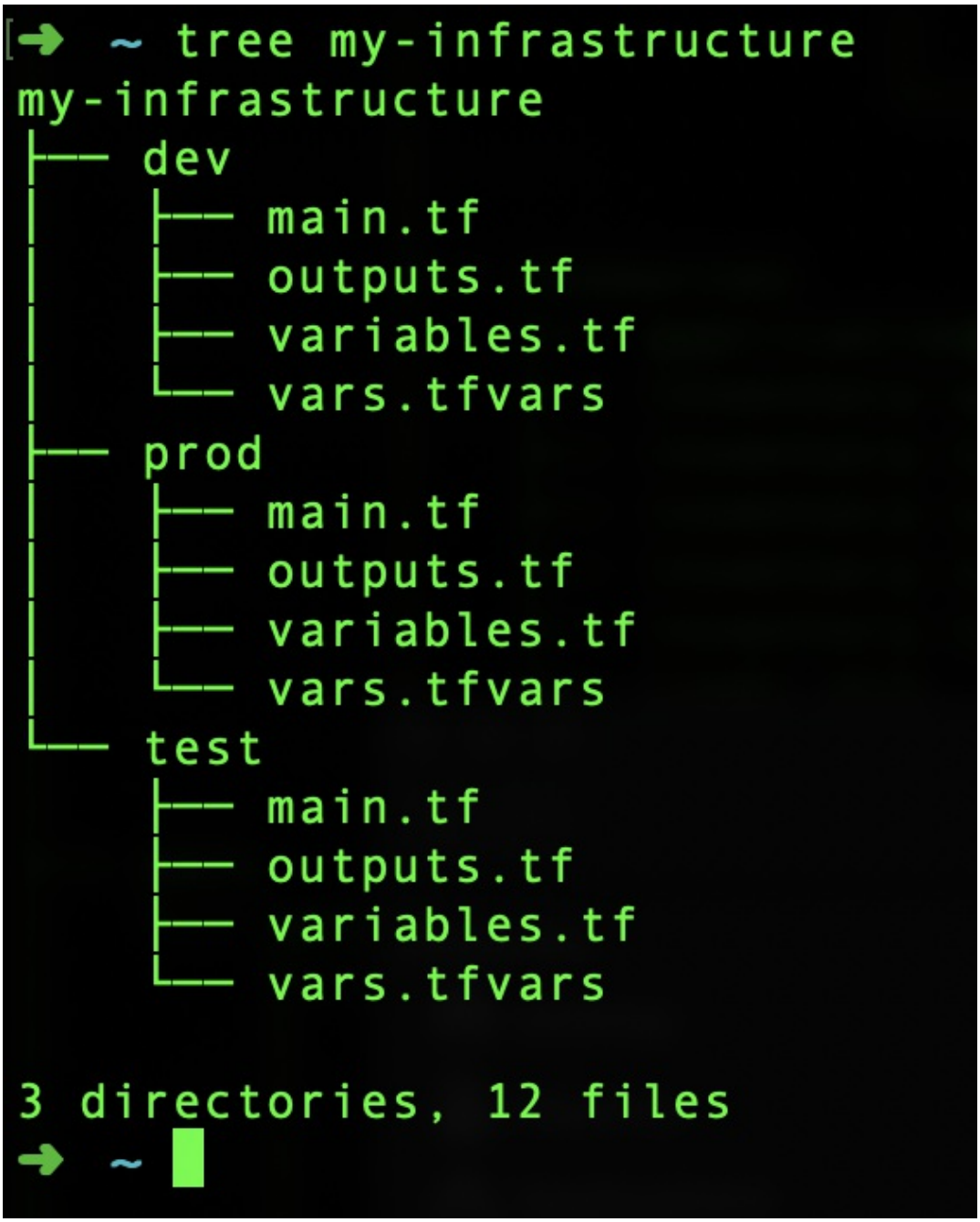
Best Practices

- Remote storage and locking, solves the problem of collaboration.
- Any problems we are left with?
    - How about Isolation?

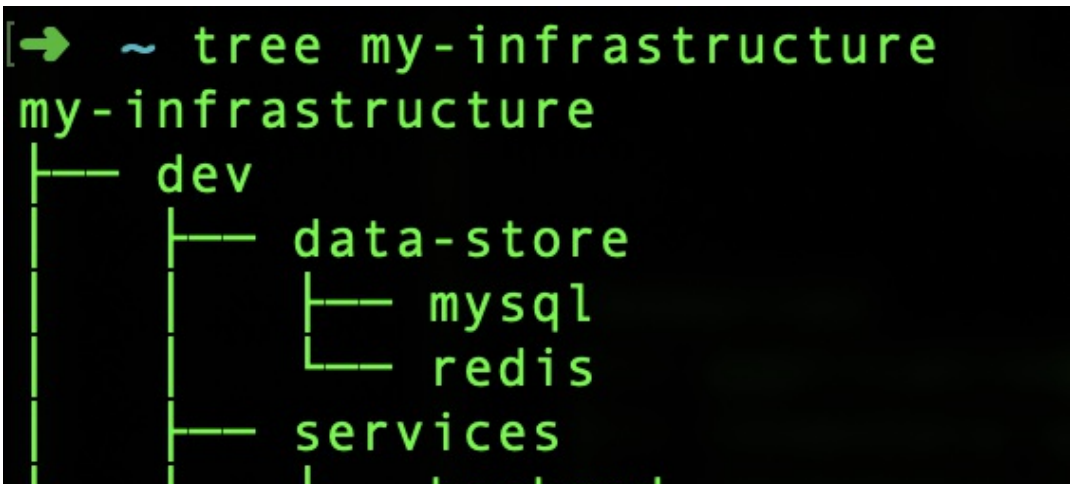
Achieving Isolation

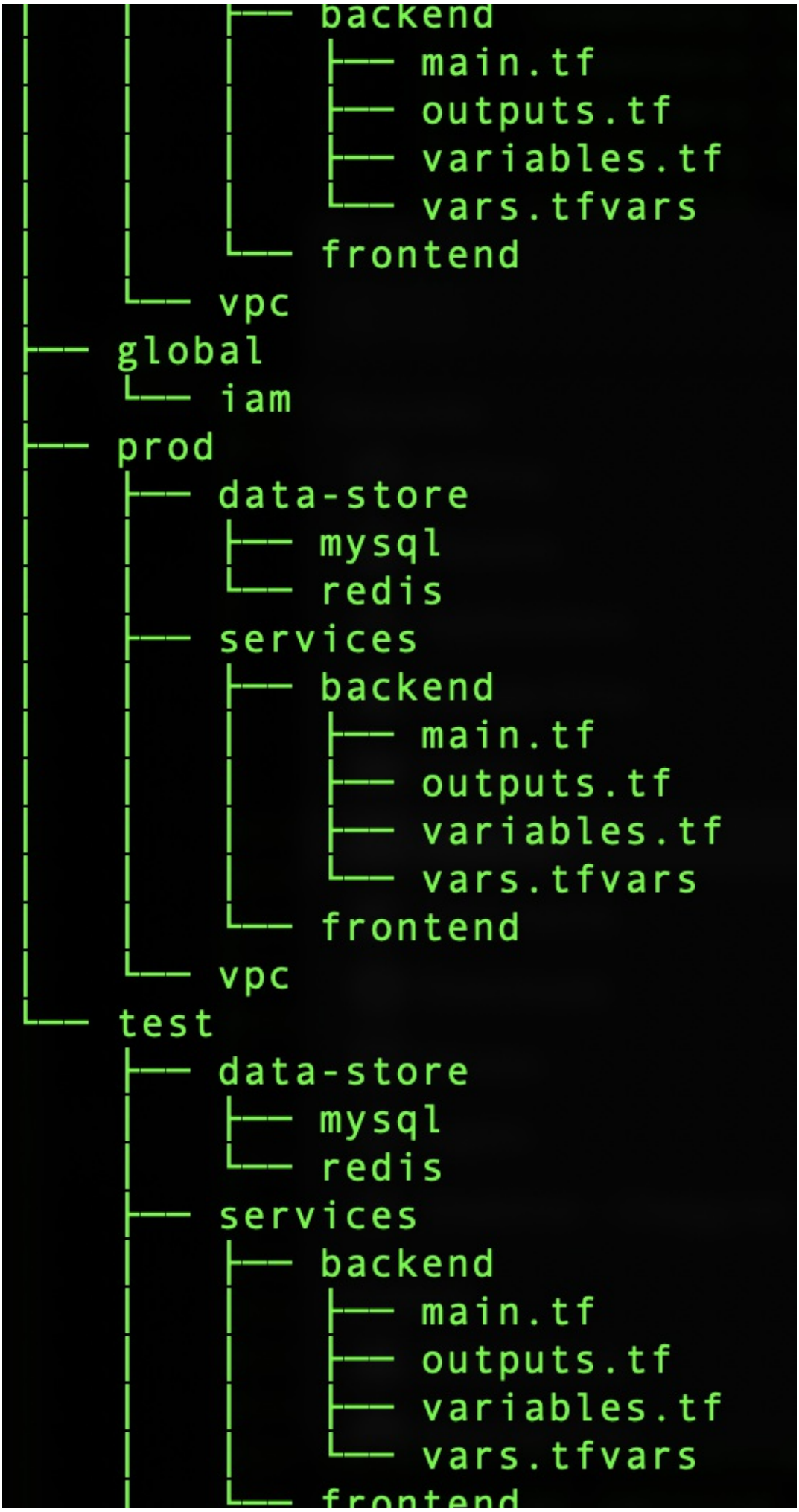
- What we have learnt so far. Define all your infra in one terraform file.
- 

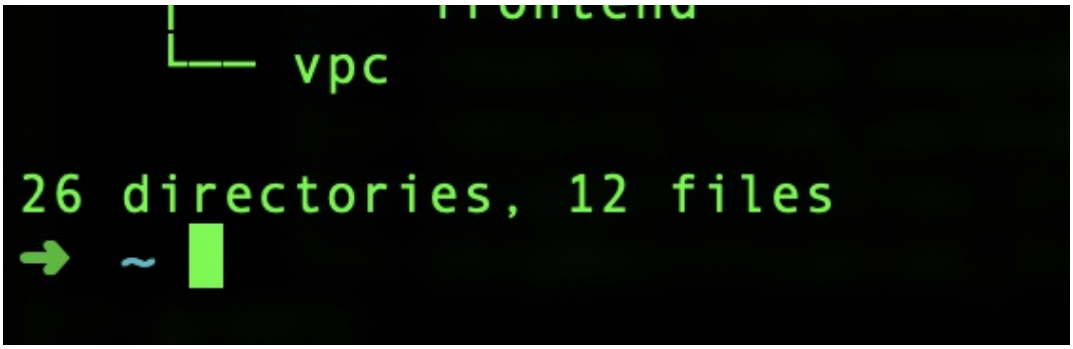
- What problems do you see?
    - Lack of using locking, might corrupt my entire state file
    - What are the chances of breaking production env while deploying something to test env?
    - The whole point of having separate environments is to isolate them from each other.

- 

- Is the above better?
- Will there be separate state files for each environment?
- Any problems you see?
  - Few things must be *global* like IAM roles.







- Any problems you see?
  - Breaking DRY principle

Read-only State

- How would you know required information of resources created as part of a different Terraform configuration (state file)?
- Or, how can you create dependent resources between two different state files?

HandsOn:

> Let's look at an example of Read-only state

> Refer 04-State/02-Remote-State-Data-Source

- Key takeaways:
  - You need to have a DynamoDB table and S3 pre-created
  - tfstate file gets created in S3 bucket. Let's verify.

> Let's destroy all resources including the s3 bucket and DynamoDB table