

Introduction to Infrastructure as Code (IaC)

Luan Truong

Traditional infrastructure deployment

Problem With Manual Management

- Prone to human error
- Time-consuming
- Difficult to track changes and maintain consistency

What exactly is Infrastructure as Code anyway?

Why should you care?

- Consistency across environments (dev, staging, prod)
- Faster deployment and scalability
- Cost savings
- Easily version-controlled via Git

IaC Tools



Here's how it works:

1. You write code describing your infrastructure.
2. Terraform says, “Cool, here's what I'm gonna do.”
3. You say “yes”, and Terraform makes it happen.

Show me the code!

```
1 provider "aws" {  
2   region = "us-west-2"  
3 }  
4  
5 resource "aws_instance" "example" {  
6   ami           = "ami-12345678"  
7   instance_type = "t2.micro"  
8 }
```


How do we deploy our Cloud infrastructure?

- **Setup everything manually!**
 - Does this scale? Clearly no.
- **Custom scripts**
 - Use your cloud provider's API to create machines
 - Programmatically SSH into the machine to do tasks
 - Does this scale? Maybe... but why reinvent the wheel?
- **Infrastructure as Code**
 - Declare your infrastructure setup in a specific format
 - Your IaC framework deploys/updates your cloud infrastructure!
 - Does this scale? Yes!

Thank you
Any questions?