ADDO
ALL DAY DEVOPS

Integrating Infrastructure as Code into a Continuous Delivery Pipeline

Considerations & Best Practices

Adarsh Shah
Technology Leader, Coach, Hands-on Architect
Independent Consultant
🐦 *@shahadarsh*
🌐 *shahadarsh.com*

We are going to cover..

- Infrastructure as Code
- Continuous Delivery
- Considerations & best practices when integrating IaC to CD
  - Source Control
  - Testing
  - Security
  - Compliance
  - Patterns for Provisioning
  - Build and Deploy pipelines
- GitOps
- People & Process

Infrastructure as Code

Infrastructure as Code (IaC) is the approach that takes **proven coding techniques** used by software systems and extends them to infrastructure.

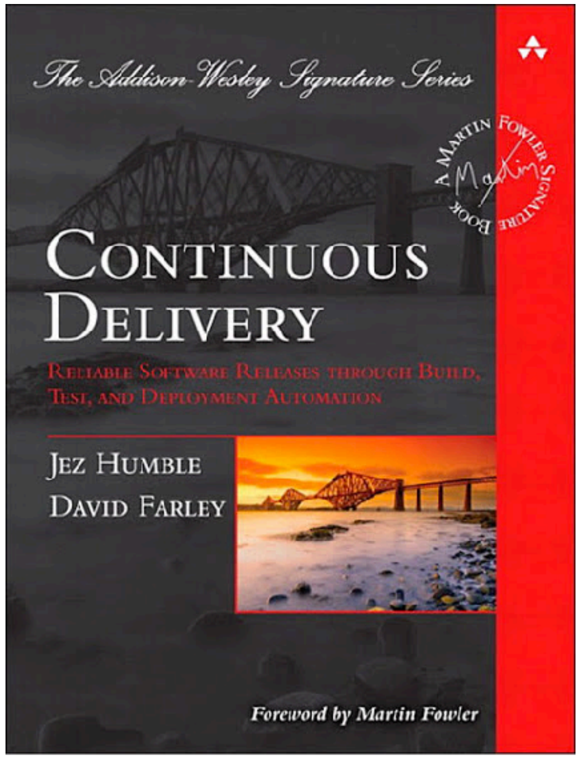
Challenges without IaC

- Configuration Drift
- Snowflake Servers
- Human Error
- Time to Complete

Continuous Delivery

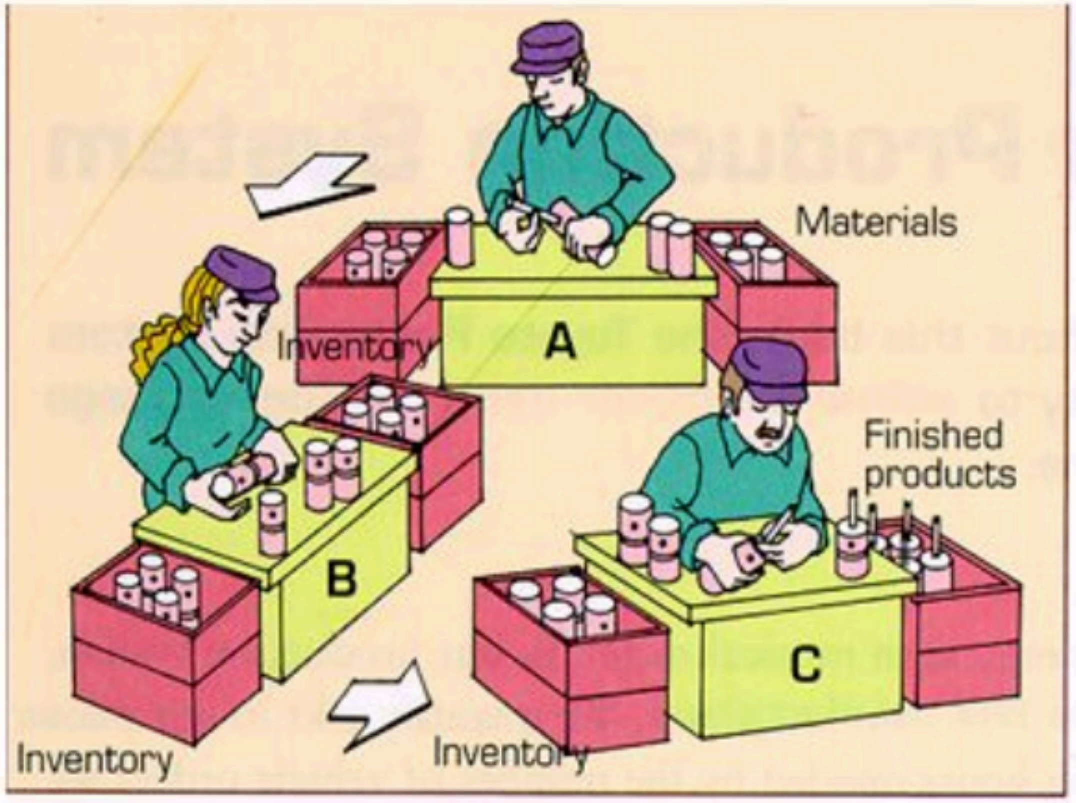
Continuous Delivery is the ability to get changes of all types—including new features, configuration changes, bug fixes and experiments—into production, or into the hands of users, safely and quickly in a sustainable way.

- Jez Humble


The Addison-Wesley Signature Series
A Martin Fowler Signature Book
Continuous Delivery
Reliable Software Releases through Build, Test, and Deployment Automation
Jez Humble
David Farley
Foreword by Martin Fowler

Continuous Delivery

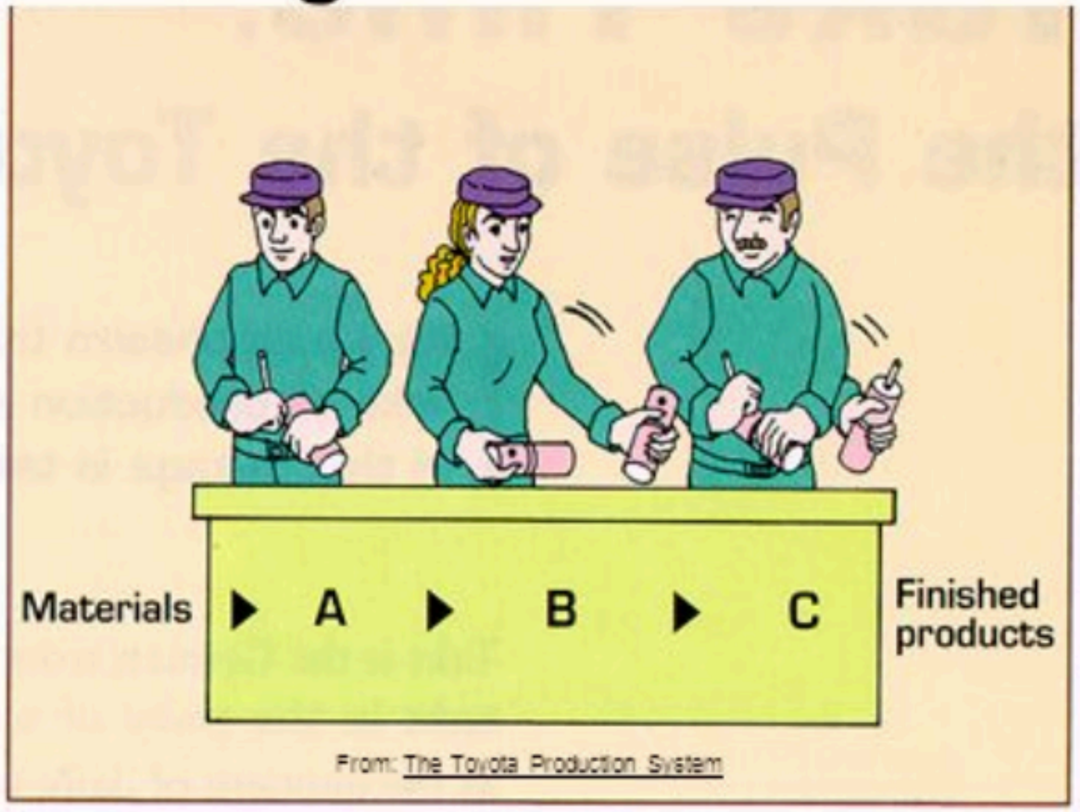
Batch



Catches Defects too Late

- How many more do you have?
- Where are they in the process?
- What is the root cause?

Single Piece Flow



From: The Toyota Production System

Catches Defects Immediately

- You only have one
- You know where it occurred
- Resolve the root cause immediately

Considerations & Best Practices

Source Control

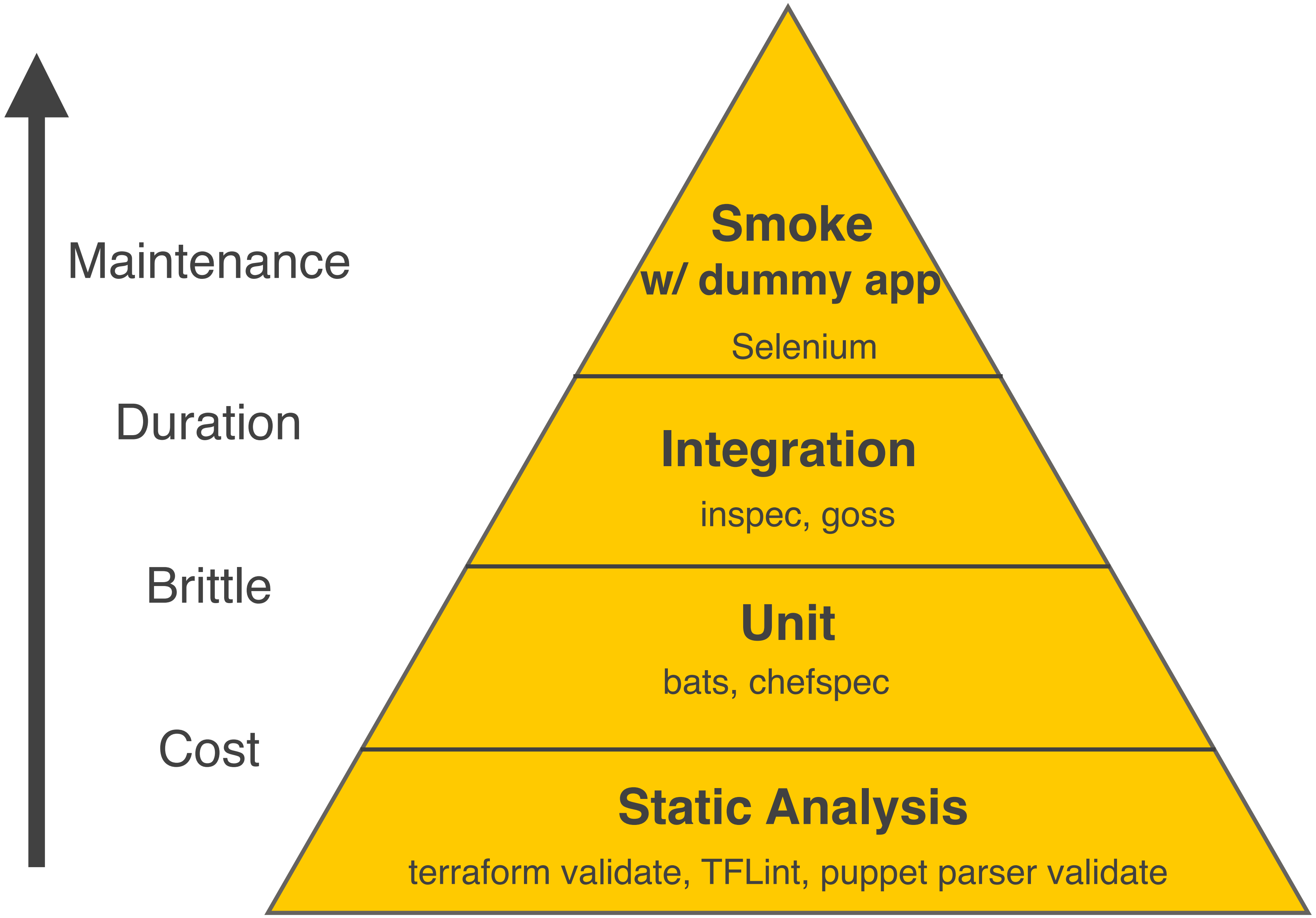
Everyone knows where to look

http://www.flickr.com/photos/thunderchild5/1330744559/

Source Control

- Everything in source control
- Code accessibility
- Modularize
    - Version the Modules
- Collaboration!!
- Code/test as documentation

Infra as Code testing



Maintenance

Duration

Brittle

Cost

Smoke w/ dummy app

Selenium

Integration

inspec, goss

Unit

bats, chefspec

Static Analysis

terraform validate, TFLint, puppet parser validate

Infra as Code Test Pyramid

Security Patterns

- CIS benchmark automation
- Building hardening policies
- Static scanning

Security Considerations

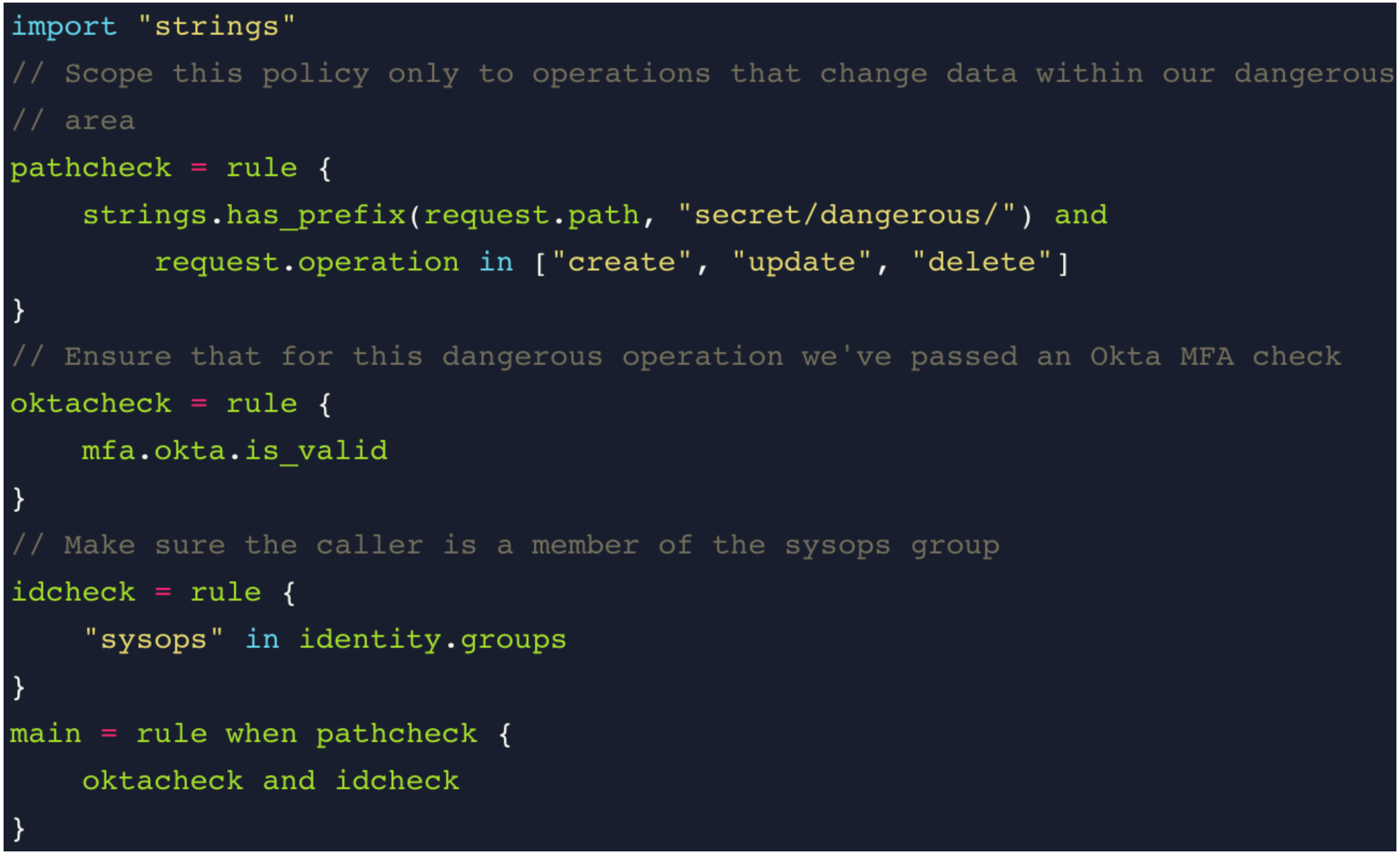
- Dynamic scanning
- Secrets management
- Artifact signing & verification

Compliance

- Finance, Healthcare & other industries
    - SOX, PII, HIPPA, PCI, GDPR
- Compliance as Code - Code instead of Paperwork
- Chef InSpec, HashiCorp Sentinel (Policy as Code)

Compliance as Code using HashiCorp Sentinel

Ensure that modification of critical data can only be performed
by authorized sysops with valid MFA



```
import "strings"
// Scope this policy only to operations that change data within our dangerous
// area
pathcheck = rule {
    strings.has_prefix(request.path, "secret/dangerous/") and
        request.operation in ["create", "update", "delete"]
}
// Ensure that for this dangerous operation we've passed an Okta MFA check
oktacheck = rule {
    mfa.okta.is_valid
}
// Make sure the caller is a member of the sysops group
idcheck = rule {
    "sysops" in identity.groups
}
main = rule when pathcheck {
    oktacheck and idcheck
}
```

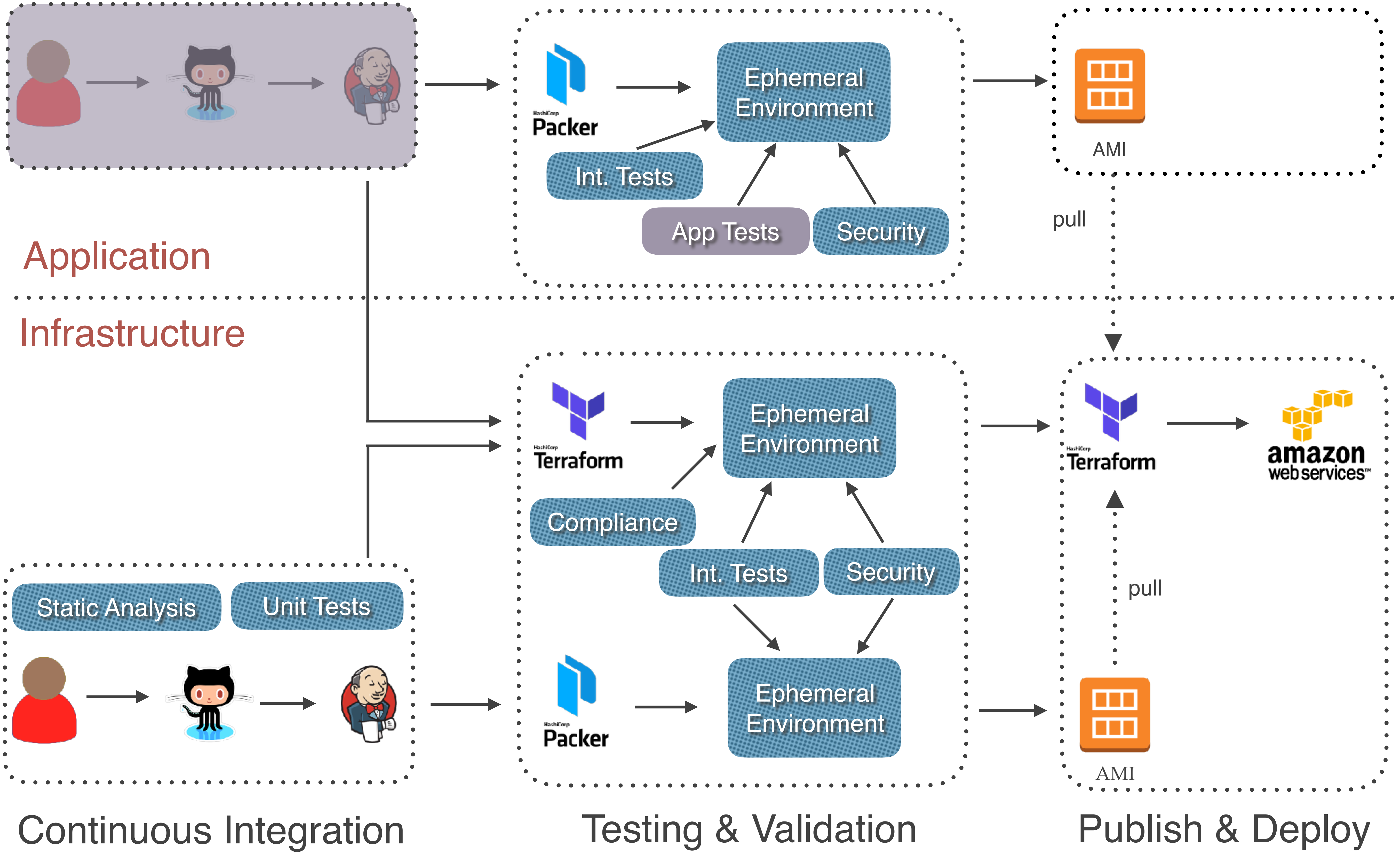
Patterns for Provisioning

- Immutable VMs
- Containerized Services
- Base Image & App Pull

Immutable VMs

- Infra Module - Multitier App
- Loosely Coupled
- App Image consumed by Infrastructure Module

Immutable VMs



Application

Packer

Ephemeral Environment

Int. Tests

App Tests

Security

AMI

pull

Infrastructure

Static Analysis

Unit Tests

Terraform

Ephemeral Environment

Compliance

Int. Tests

Security

Packer

Ephemeral Environment

Terraform

amazon web services

pull

AMI

Continuous Integration

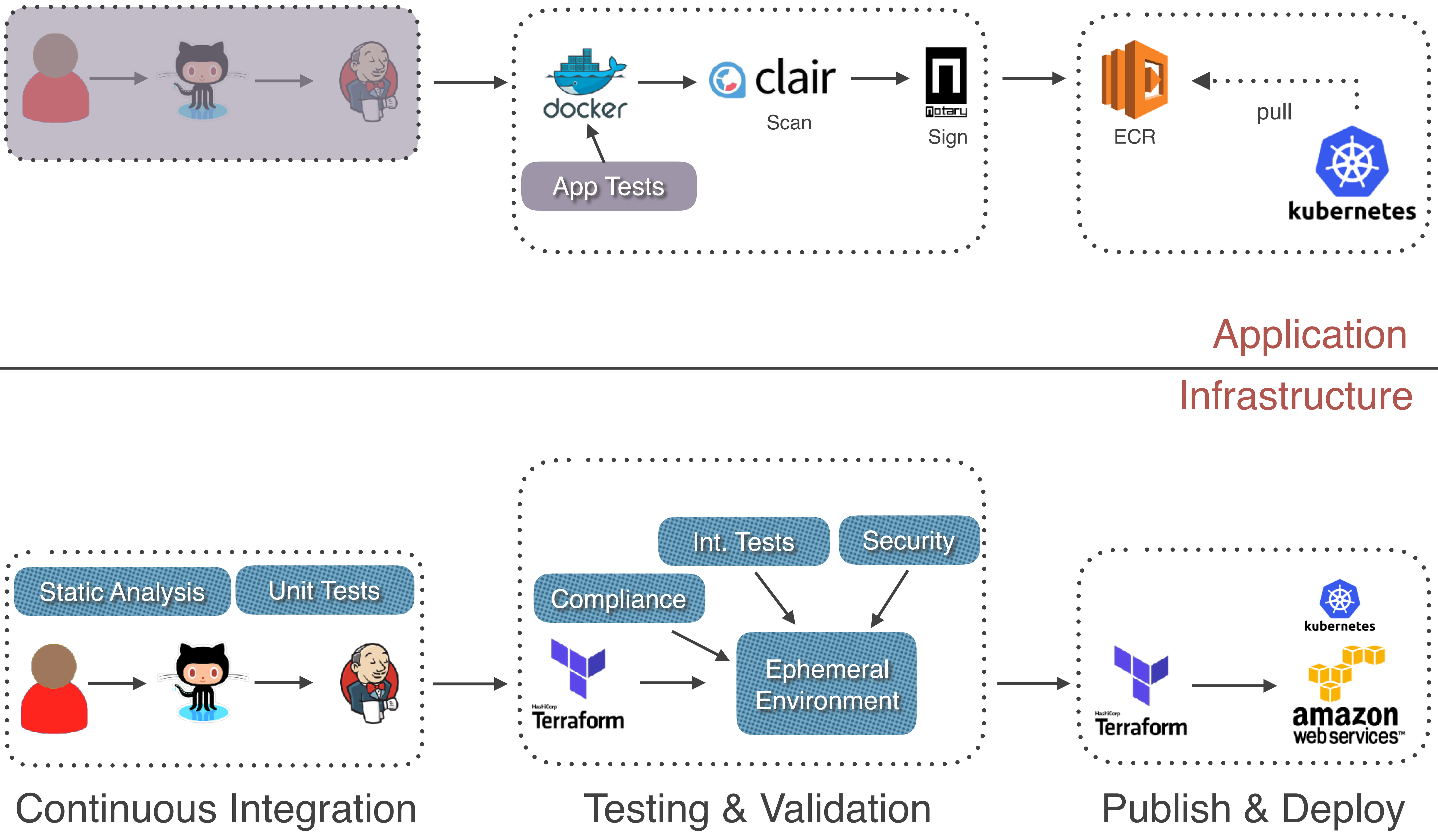
Testing & Validation

Publish & Deploy

Containerized Services

- Infra Module - Container Management System
- Fully Decoupled from Apps
- Apps are deployed with Container Management System specific tools

Containerized Services



docker

clair

Scan

Notary

Sign

App Tests

ECR

pull

kubernetes

Application

Infrastructure

Static Analysis

Unit Tests

Compliance

Int. Tests

Security

Ephemeral Environment

HashiCorp Terraform

kubernetes

amazon web services™

Continuous Integration

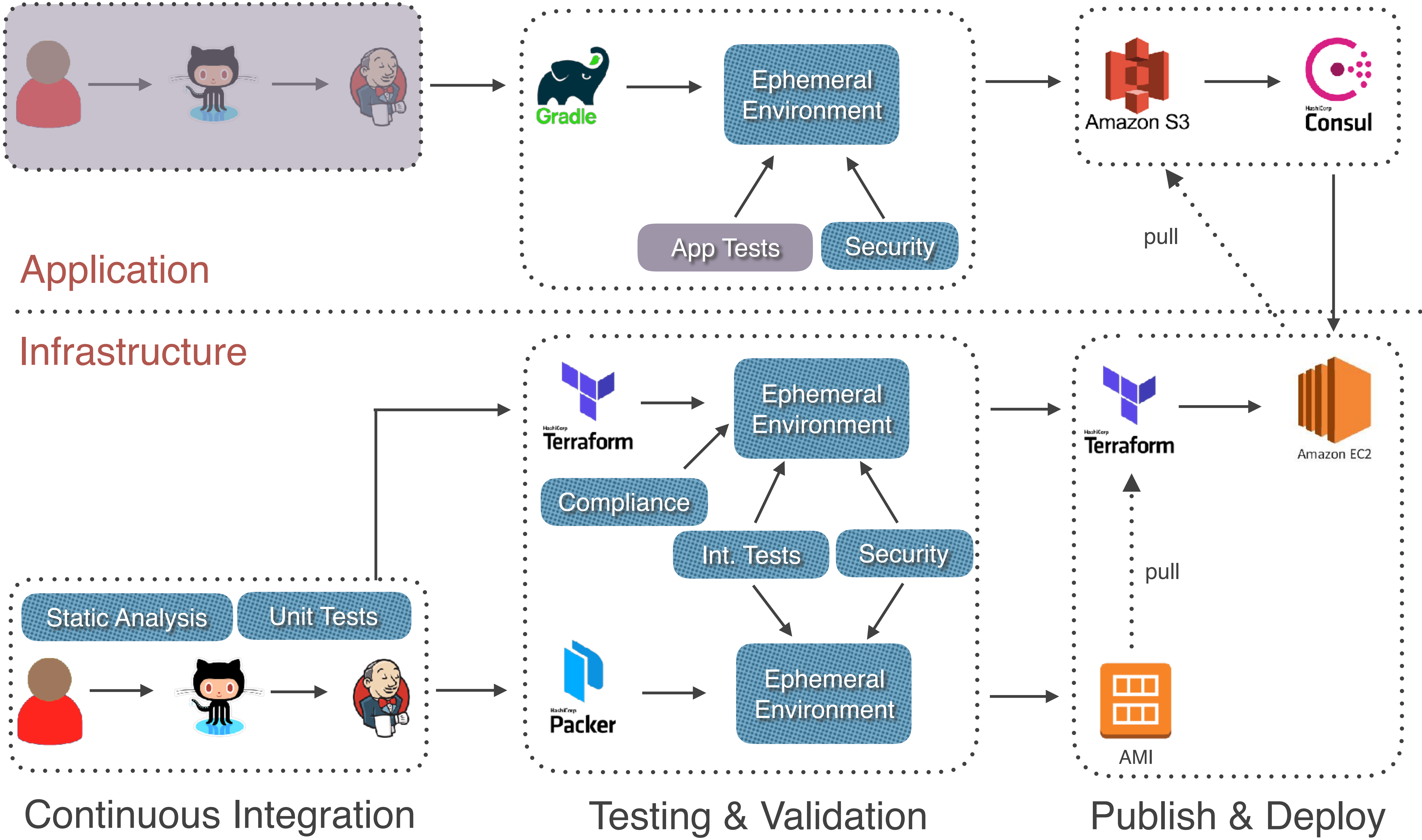
Testing & Validation

Publish & Deploy

Base Image & App Pull

- Infra Module - App Servers
- VMs pull app on deploy, or app update
- **Anti-Pattern**: Allowing Long-Lived VMs

Base Image & App Pull



Application

Infrastructure

Continuous Integration

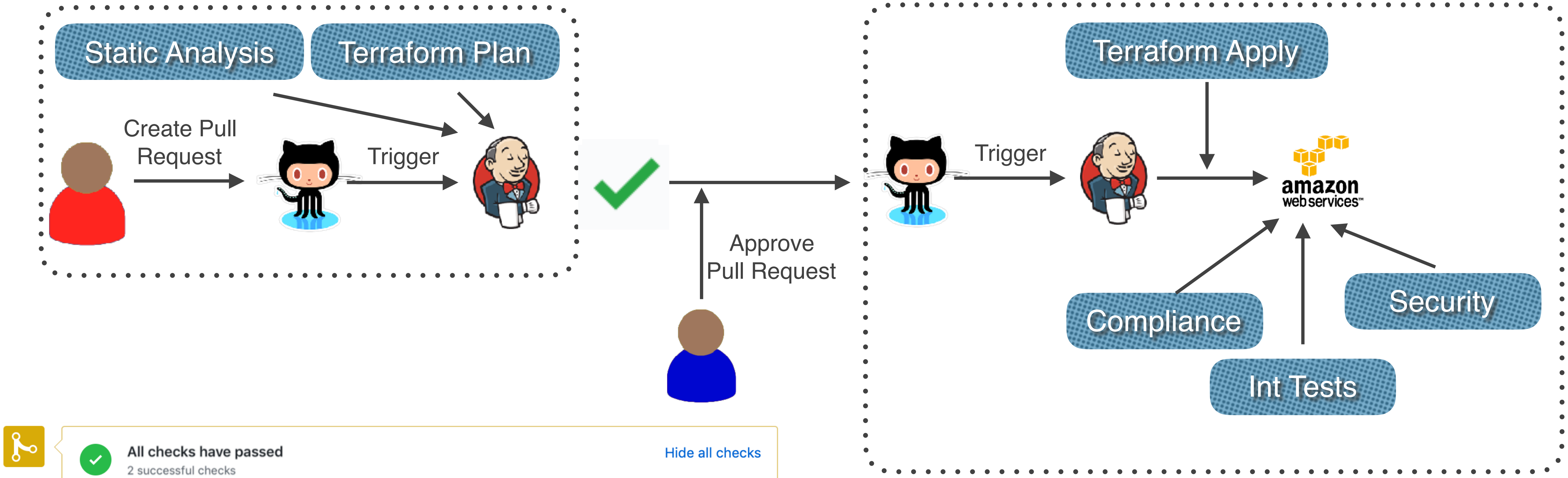
Testing & Validation

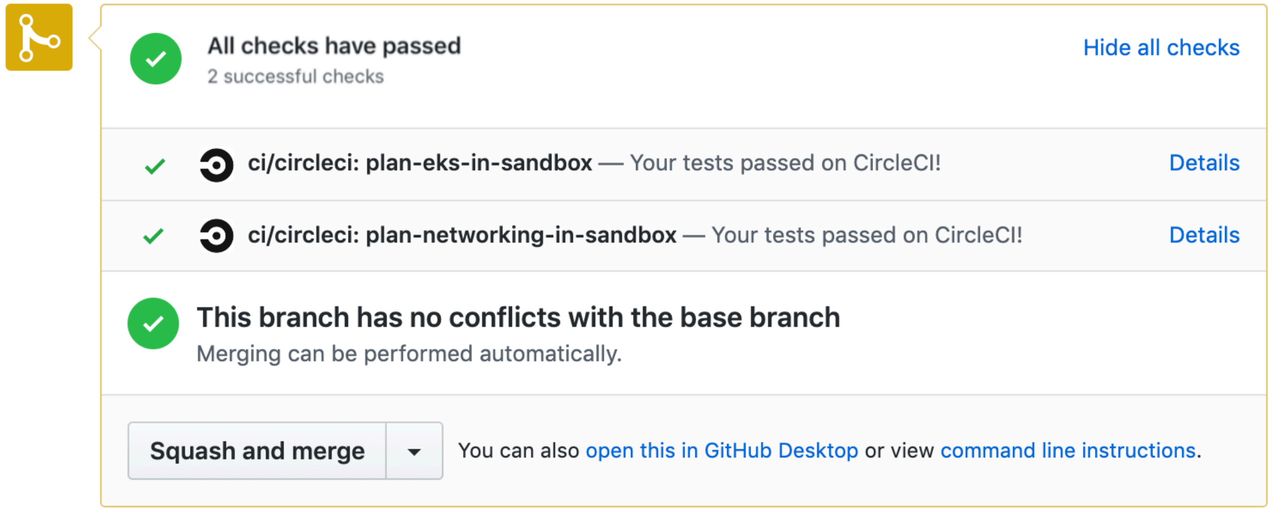
Publish & Deploy

GitOps

- Source Control as a System of Record
- Operations by Pull Request
- Repeatable, Predictable, Auditable & Accessible

GitOps Workflow





All checks have passed
2 successful checks

✓ ci/circleci: plan-eks-in-sandbox — Your tests passed on CircleCI! Details

✓ ci/circleci: plan-networking-in-sandbox — Your tests passed on CircleCI! Details

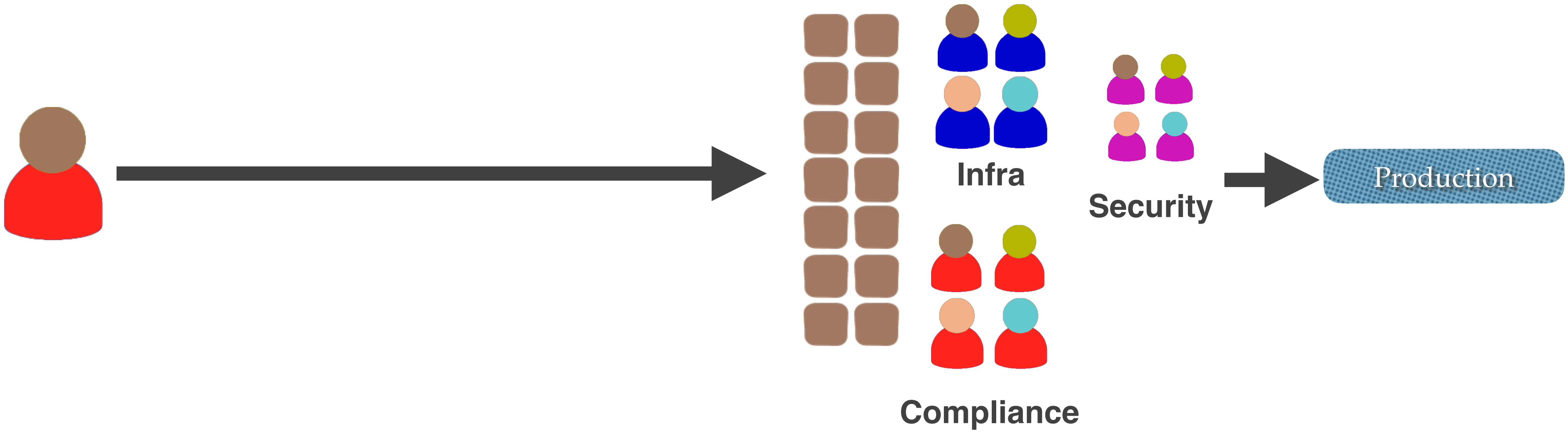
This branch has no conflicts with the base branch
Merging can be performed automatically.

You can also open this in GitHub Desktop or view command line instructions.

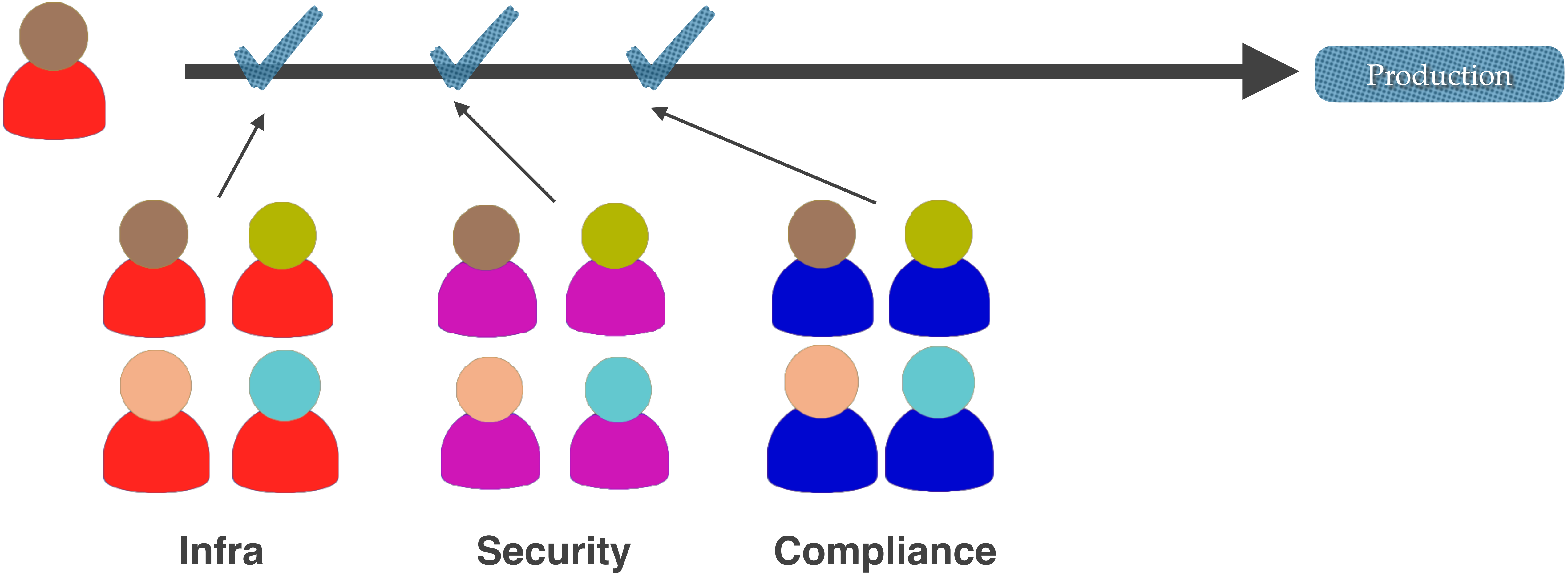
People & Process

- Enables teams to interact
- Infra, Security, Compliance, QA etc teams work together
- Improvement in processes
- Faster feedback

Inspection



Building Quality In



Summary

- Infrastructure as Code
- Continuous Delivery
- Considerations & best practices when integrating IaC to CD
    - Source Control
    - Testing
    - Security
    - Compliance
    - Patterns for Provisioning
    - Build and Deploy pipelines
- GitOps
- People & Process

ADDO
ALL DAY DEVOPS

Questions on
Slack #2019addo-ci-cd

Adarsh Shah
Technology Leader, Coach, Hands-on Architect
Independent Consultant
🐦*@shahadarsh*
🌐*shahadarsh.com*