

Continuous Delivery Using Jenkins



Justin Menga

FULL STACK TECHNOLOGIST

@jmenga pseudo.co.de

Introduction

Continuous Delivery using Jenkins

- Test, Build and Release using the Pipeline Plugin
- Jenkins in a Container
- Configure and test our workflow
- Deploy Jenkins to AWS EC2 Container Service using CloudFormation
- Github and Docker Hub integration

Demo

Setting up Jenkins Locally

- Creating a Jenkins Image
- Configuring the Workflow
- Testing the Workflow
- Creating a Self Defined Workflow
- Test Failures

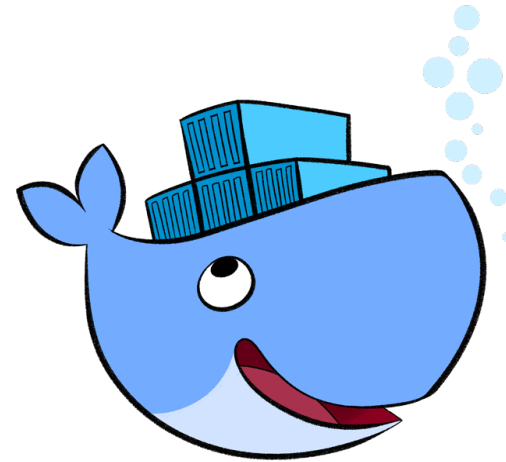
Creating a Jenkins Docker Image

Jenkins Continuous Delivery Image



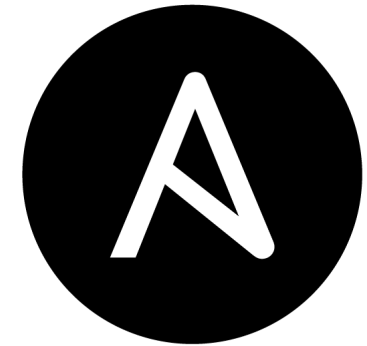
Official Jenkins Image

+

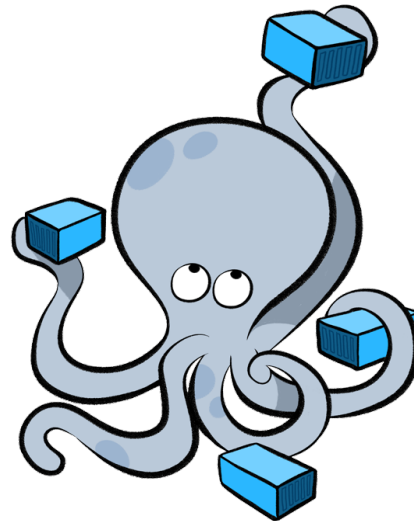


Docker

+



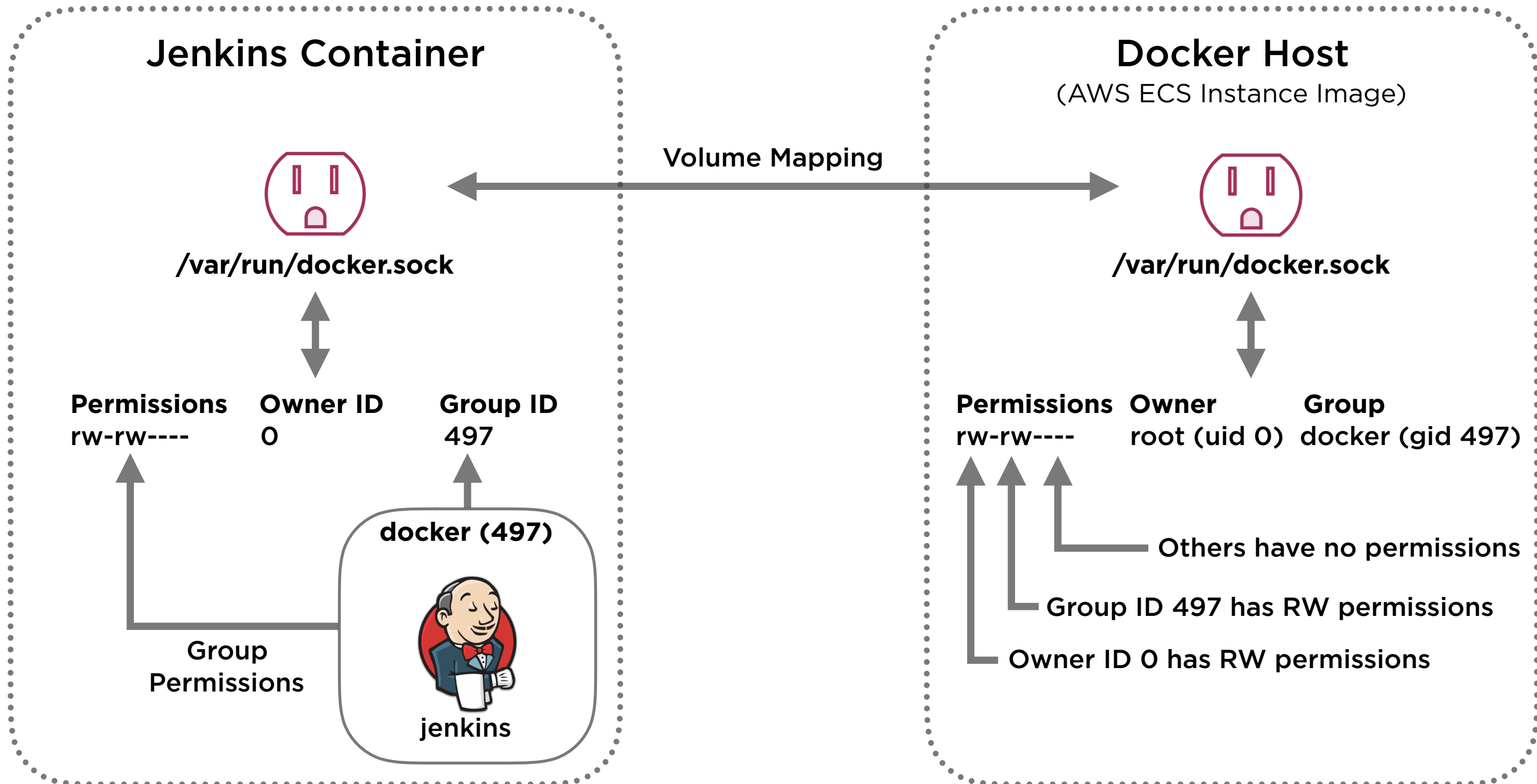
ANSIBLE



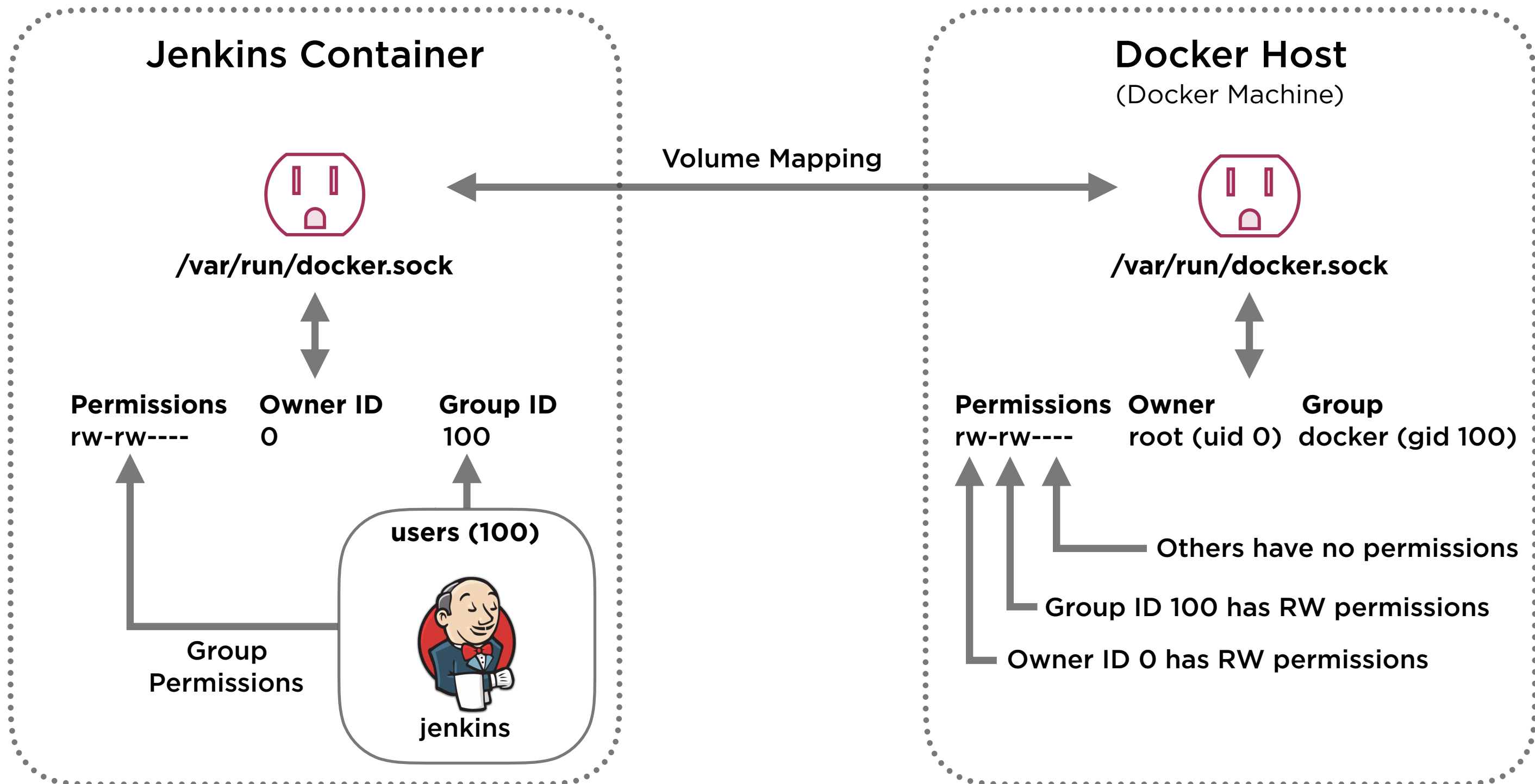
Docker Compose



Sharing the Docker Socket File



Docker Socket File on Docker Machine



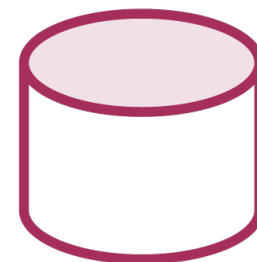
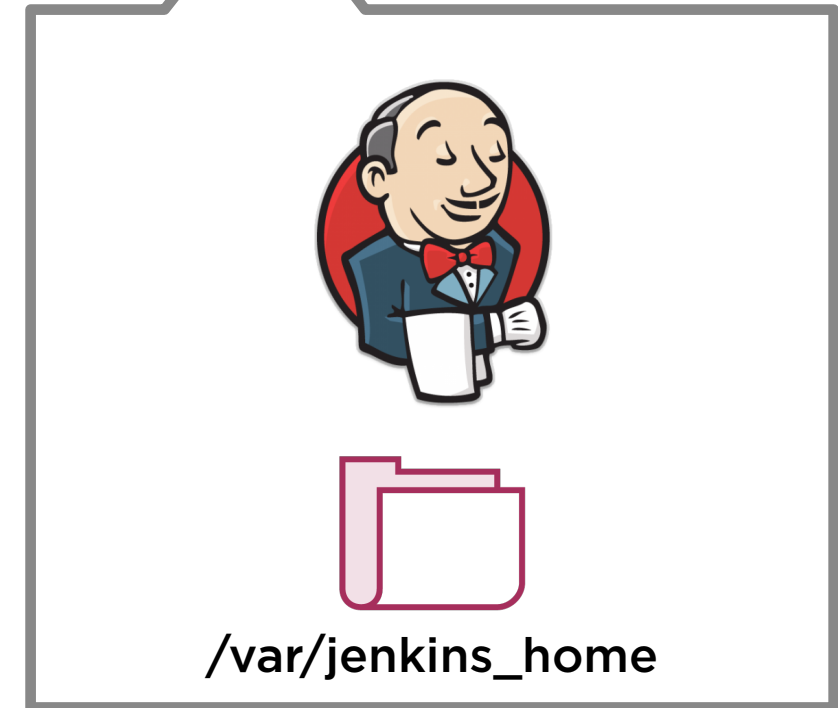
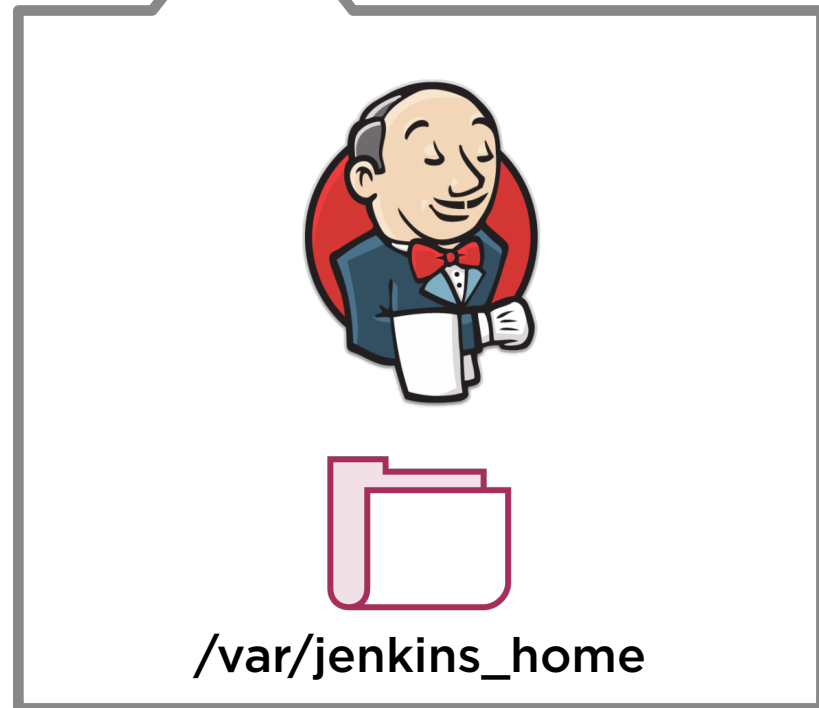
Persisting Jenkins Data



Create

Stop

Start



jenkins_home

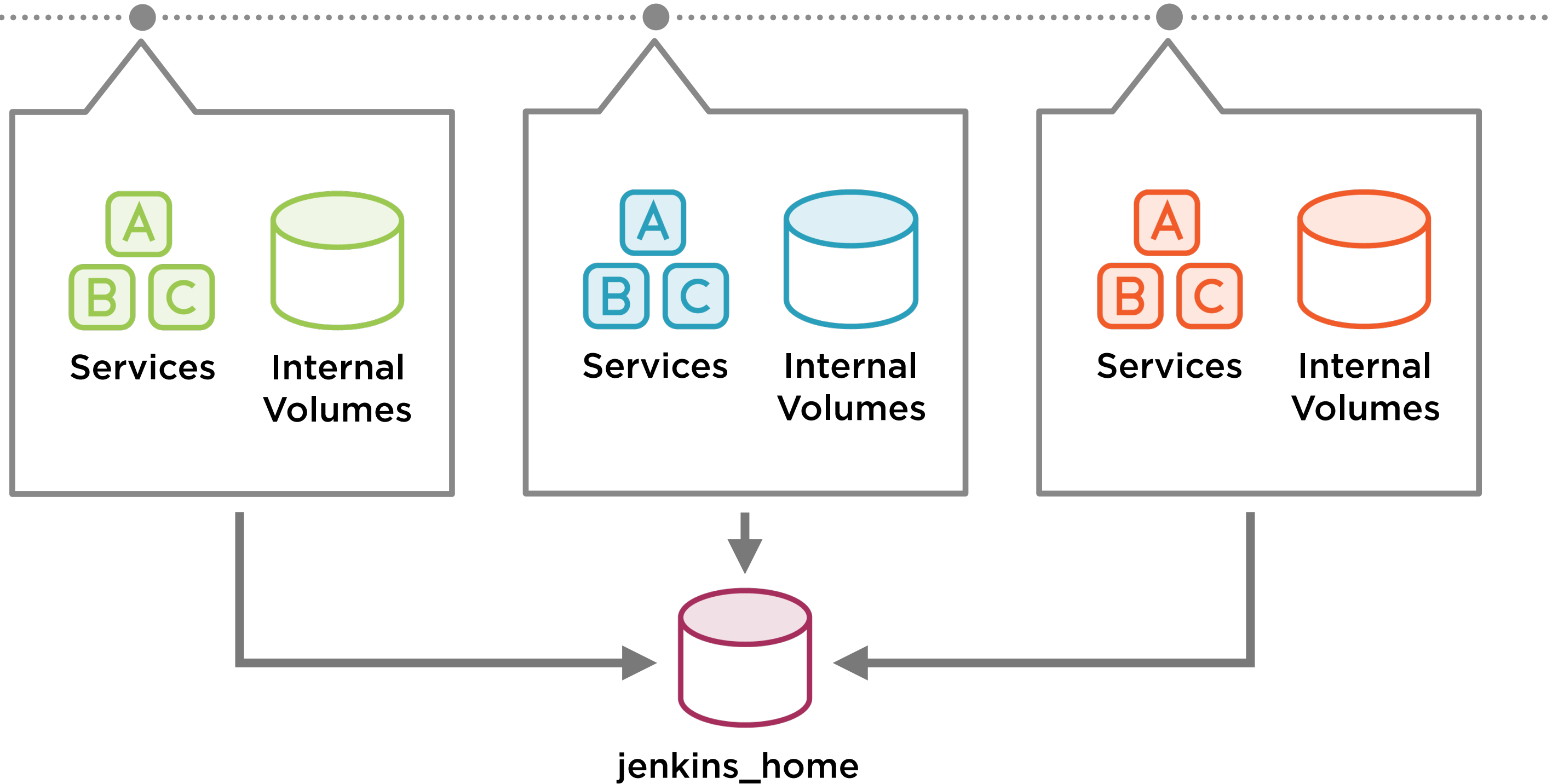
Persisting Jenkins Data



Create

Stop

Create



Configuring the Workflow in Jenkins

Testing the Workflow

Creating a Self Defined Workflow

Self Defined Workflow

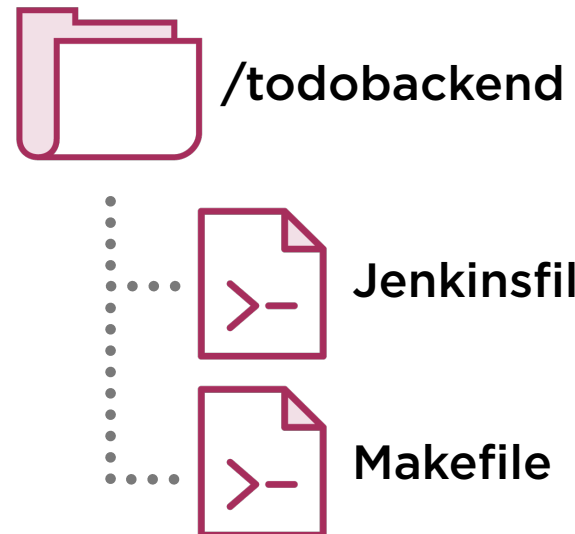
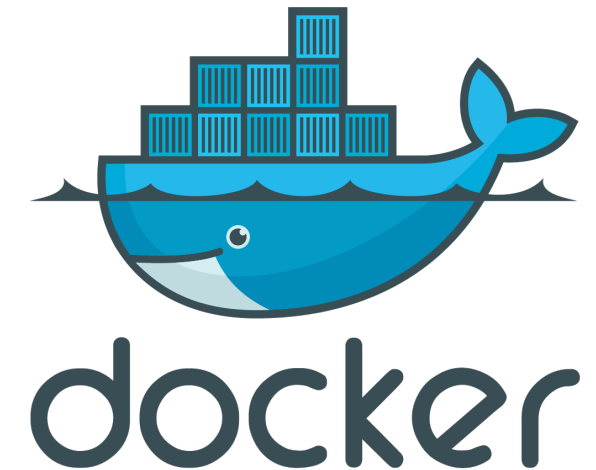
GitHub



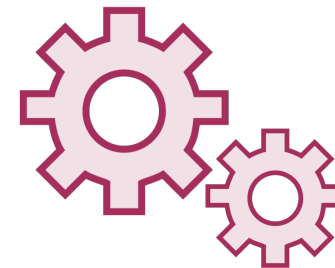
Jenkins



Docker Hub



1. Git Checkout



2. Run Workflow from Jenkinsfile

3. Publish Image

jmenga/todobackend

Test Failures

Publishing the Jenkins Image

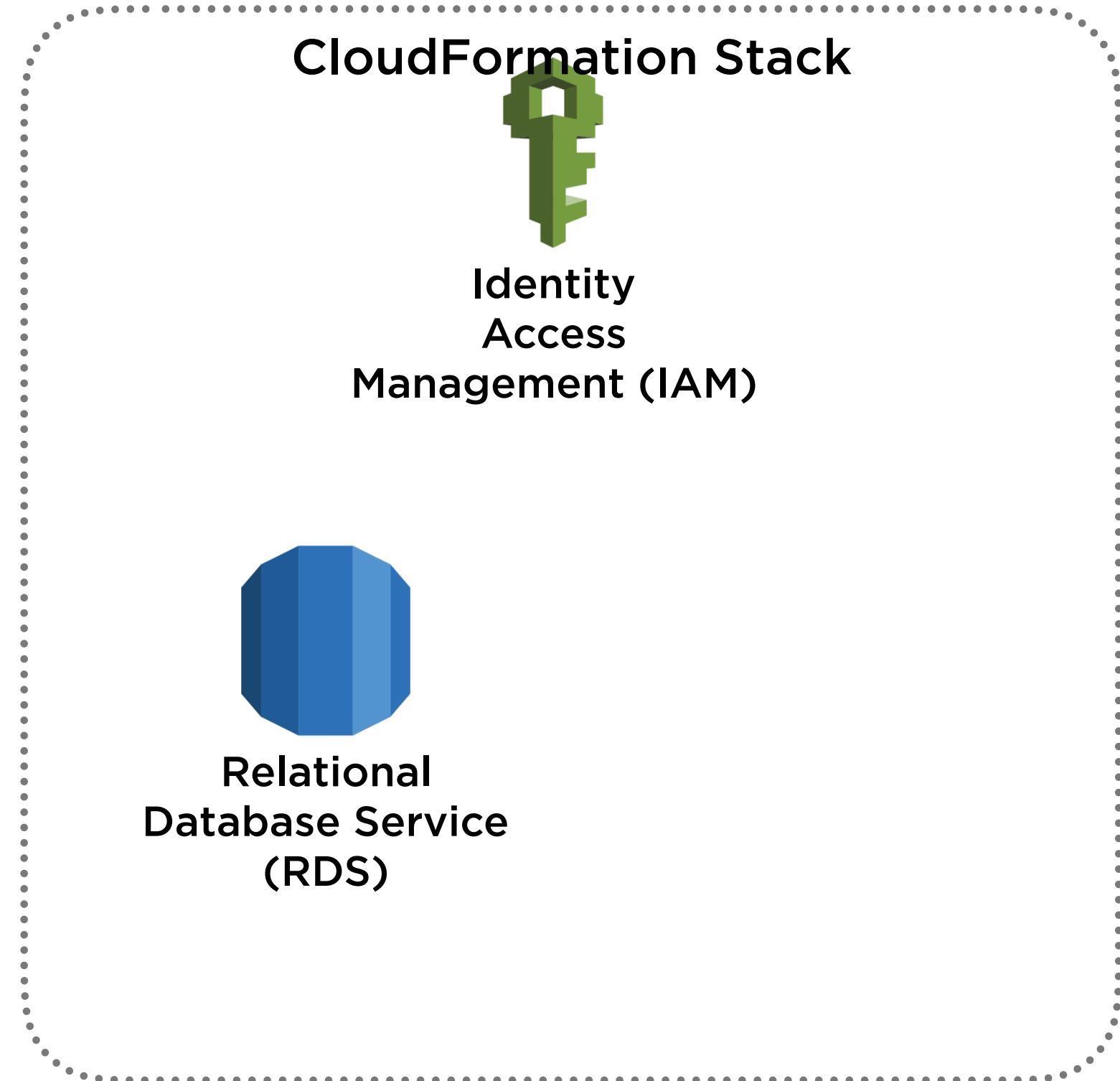
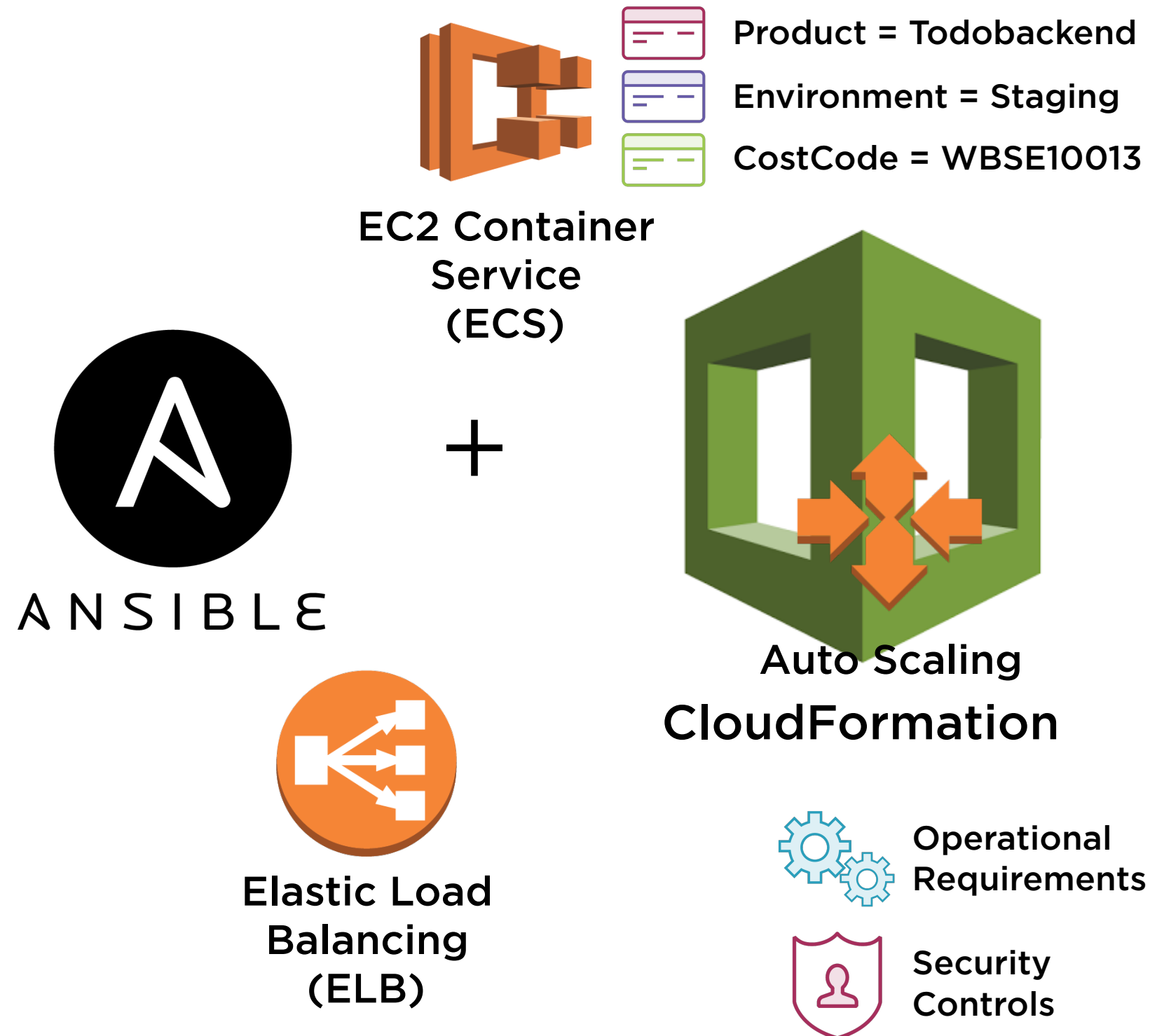
Demo

Deploying Jenkins to AWS

- AWS Deployment Strategy
- Initial AWS Setup
- AWS VPC Infrastructure
- Creating a CloudFormation Template
- Deploying to AWS

AWS Deployment Strategy

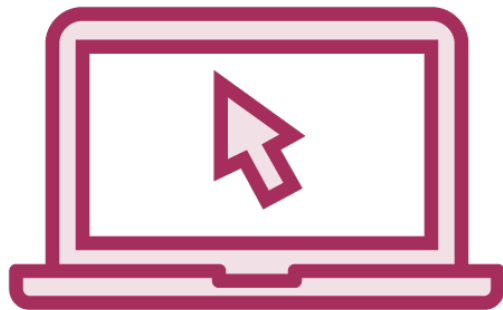
Key Deployment Technologies



Initial AWS Setup

EC2 Key Pair

Management Host



EC2 Instance



EC2 Key Pair

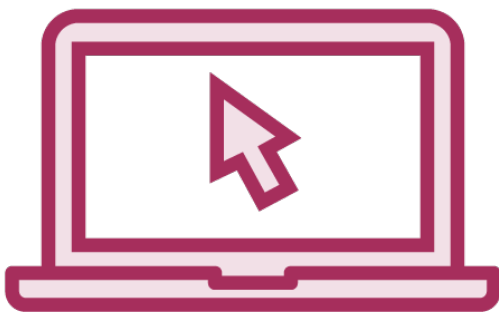


Private Key

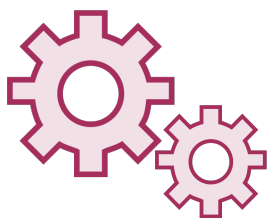
Public Key

EC2 Key Pair

Management Host

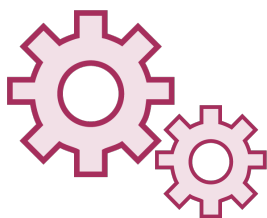


EC2 Instance

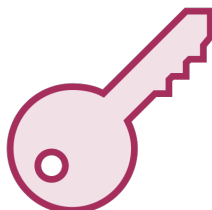


Private Key

Secure Shell (SSH)



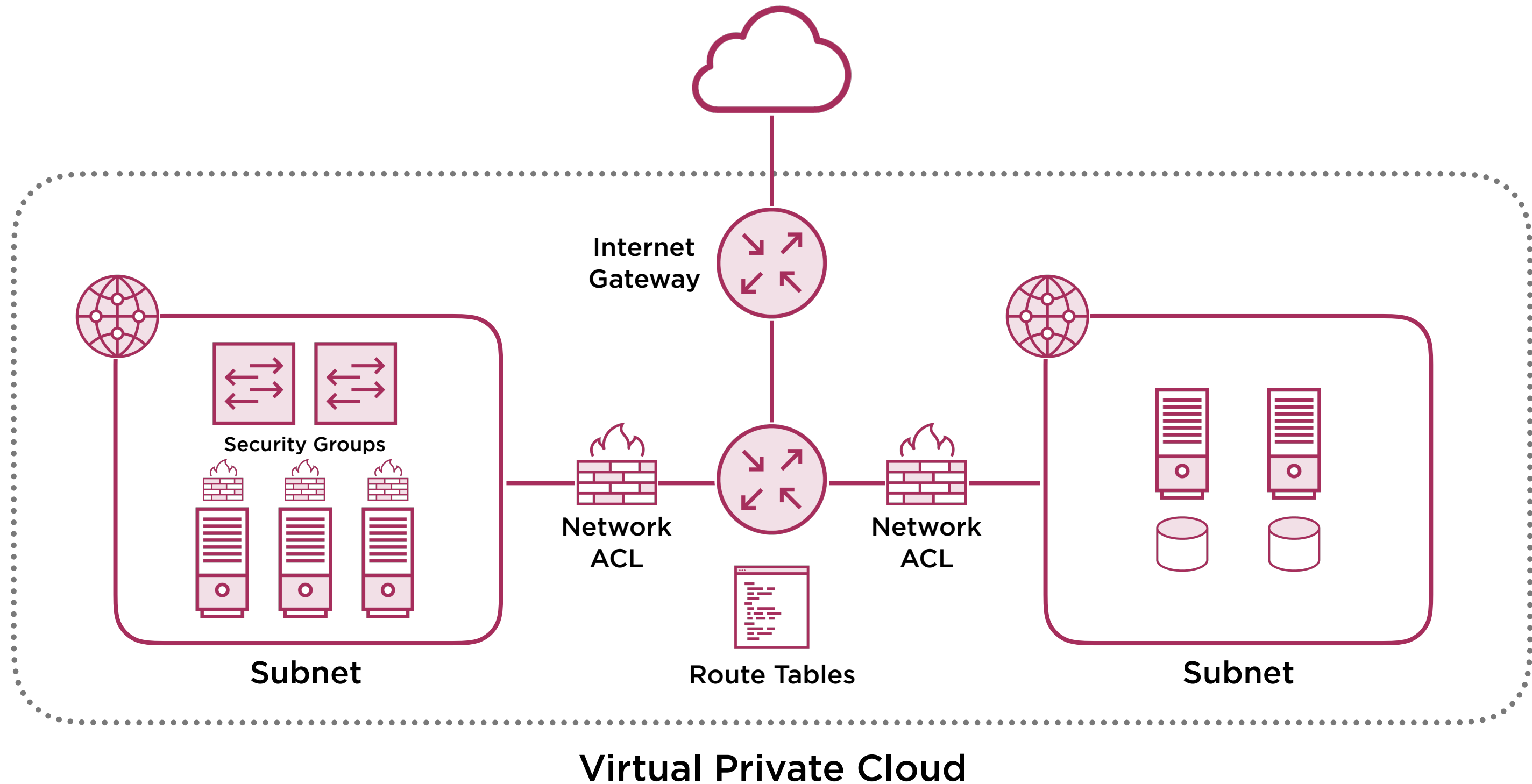
RSA Public Key
Authentication



Public Key

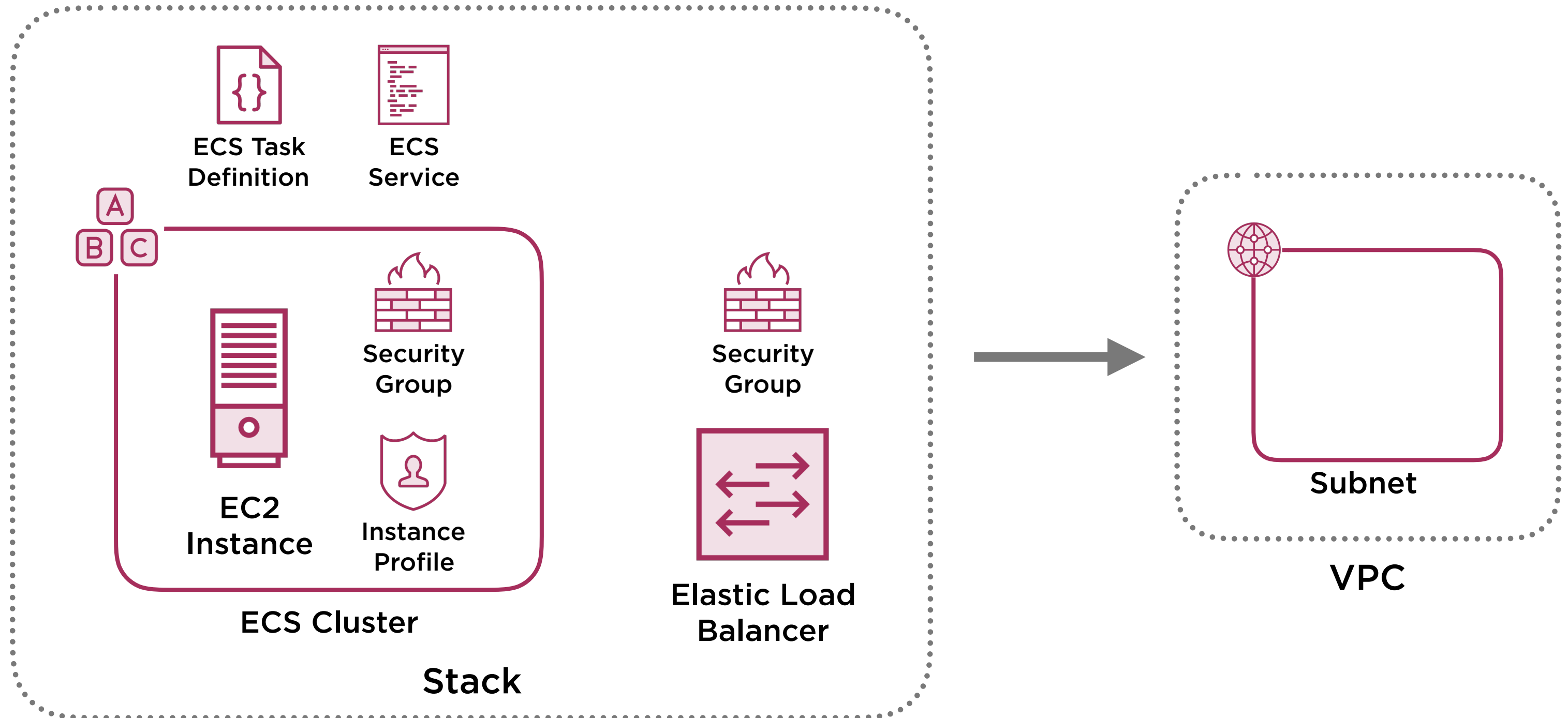
AWS VPC Infrastructure

AWS VPC Topology



Creating a CloudFormation Template

Jenkins CloudFormation Stack



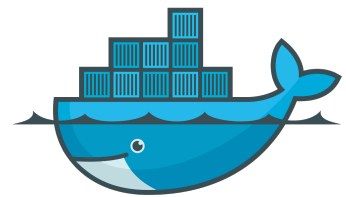
Defining an EC2 Instance

Defining Security Groups and Roles

Defining Elastic Load Balancers

Elastic Load Balancer Topology

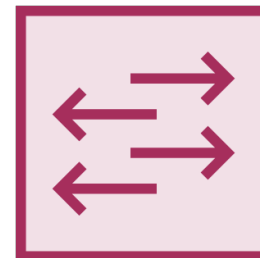
Docker Hub



docker



HTTP
Port 80



Elastic Load
Balancer

HTTP
Port 8080



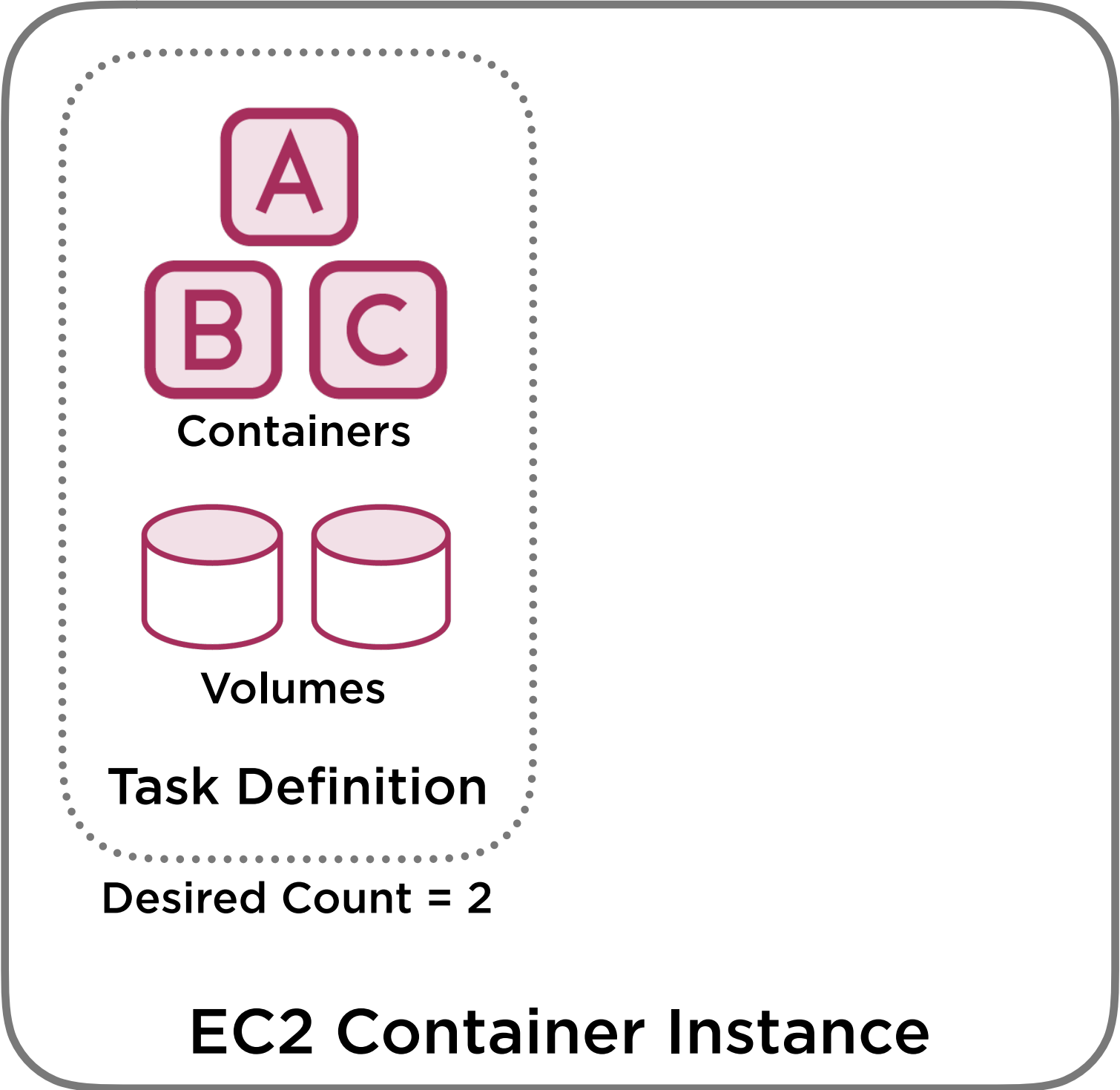
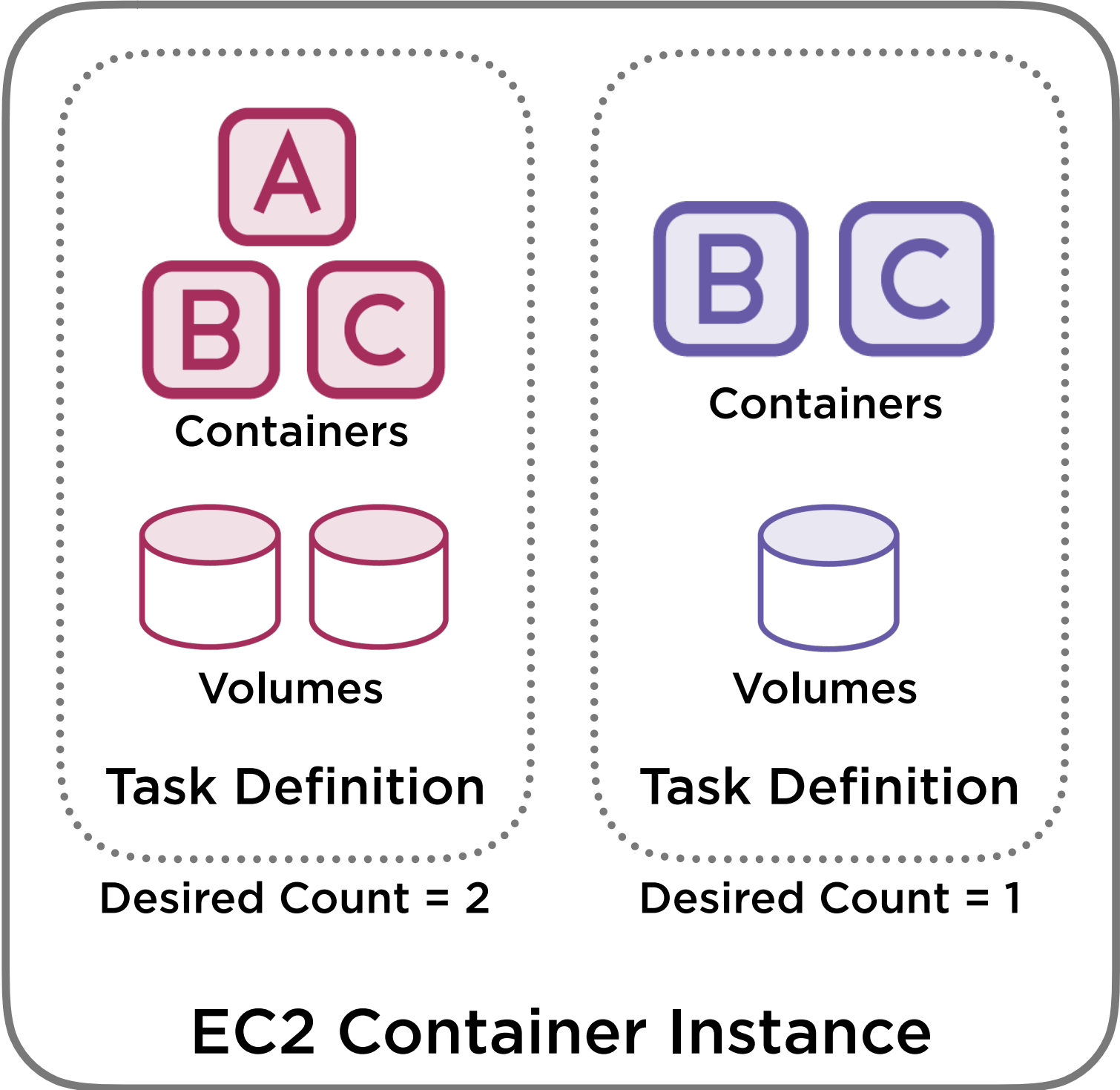
Jenkins EC2
Instance



GitHub

Configuring the EC2 Container Service

ECS Task Definition Deployment



Configuring Stack Outputs

Creating the Stack

Demo

Integrating Jenkins with GitHub and Docker Hub

- Securing Jenkins
- Testing the Workflow
- Configuring GitHub integration
- Configuring Docker Hub integration

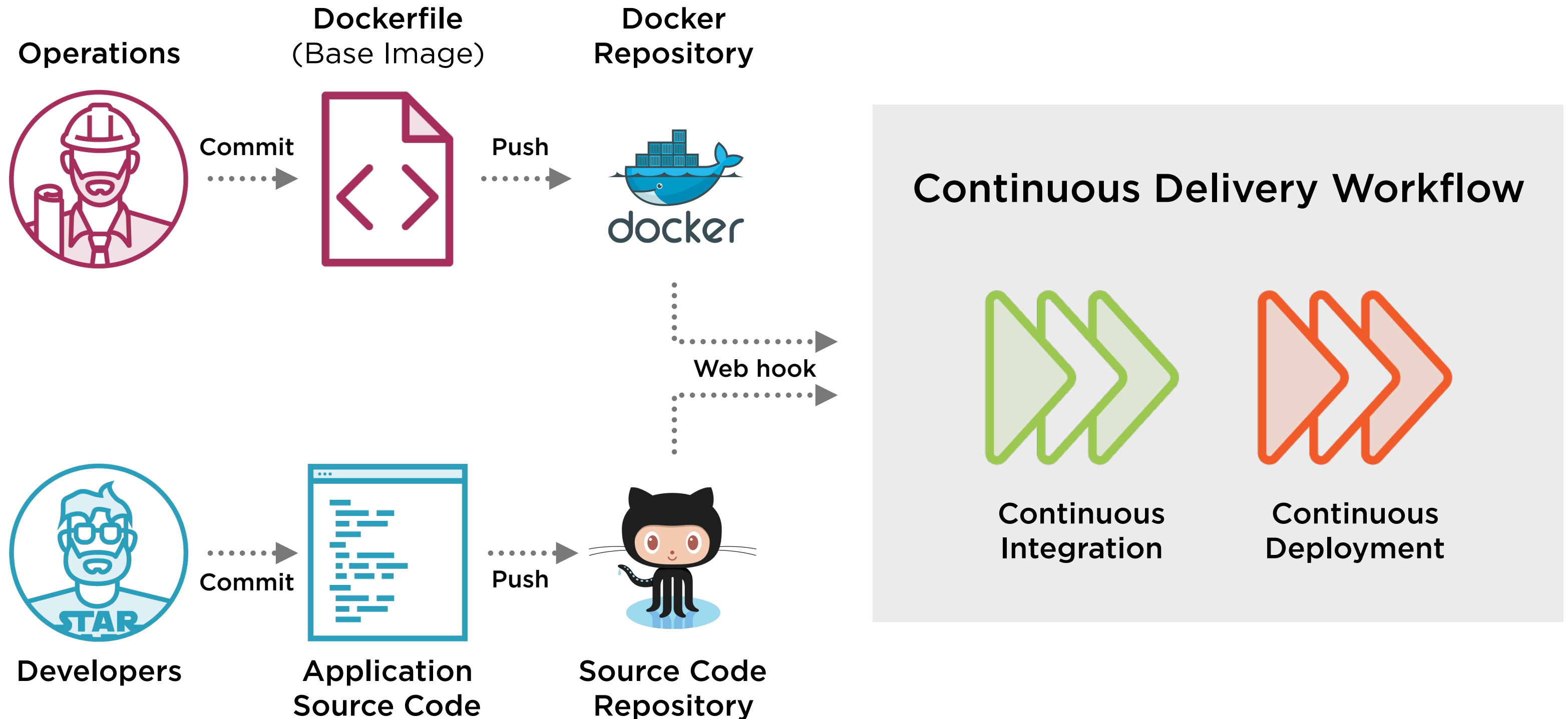
Securing Jenkins

Testing the Workflow

Configuring GitHub Integration

Configuring Docker Hub Integration

Continuous Delivery Workflow Triggers



Summary

Jenkins

- Created a Jenkins image
- Added build arguments for different environments
- Pipeline plugin

Jenkins in AWS

- CloudFormation stack
- GitHub and Docker Hub integration