

# Automating Deployments with Azure DevOps and GitHub

---



**Marcel de Vries**

CTO

@marcelv

<https://fluentbytes.com>



# Outline



Secure and compliant CI/CD

Security scanning

Deploying to AKS using Azure DevOps

Deploying to AKS using GitHub Actions

Summary



# Secure and Compliant CI/CD



## Security considerations for CI/CD

- Scan container images as part of your build process
- Ensure secrets are managed in an appropriate way in the pipeline
- Ensure secrets are managed in the cluster
  - E.g. connection strings, tokens, etc.

# Security Scanning



**Scanning can be done at multiple places**

- Azure Security Center
- As part of the container registry
- Step in the CI/CD pipeline

**When vulnerability found it should block further publication to registry**

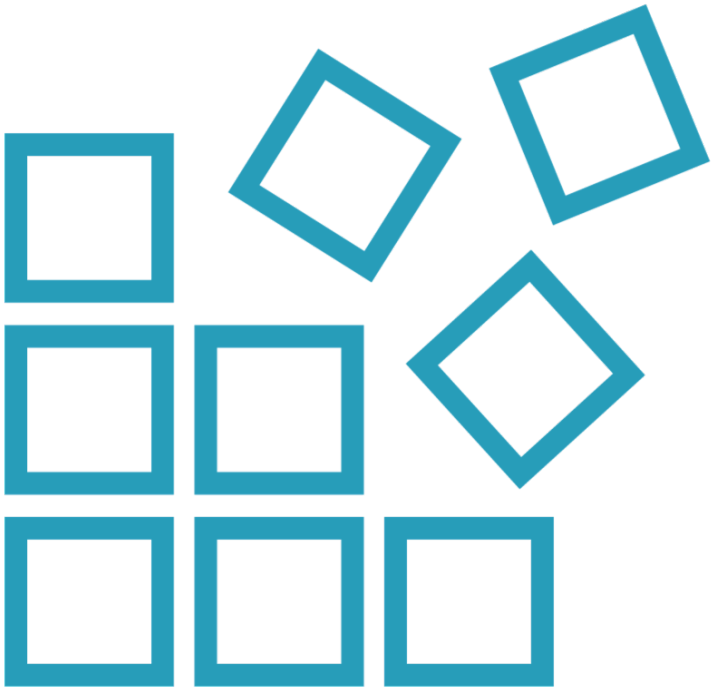
# Secure and Compliant CI/CD



## Compliance considerations for CI/CD

- Ensure you have a 4 eyes principle
  - E.g. Pull requests on every change
- Ensure you have traceability
  - E.g. use GitHub Issues or Azure Boards

# Create a Pipeline per Microservice



**You want each microservice to be independent**

**Each change is deployed independently**

**Each microservice is independently versioned**

**Each pipeline can have specific requirements to ensure compliance**

- E.g. Public web facing vs. internal

# Demo



## Deploying to AKS Using Azure DevOps



# Demo



## Deploying to AKS Using GitHub Actions





# Summary



**Secure and compliant CI/CD**

**Security scanning**

**Deploying to AKS using Azure DevOps**

**Deploying to AKS using GitHub Actions**

