Uma Abordagem DevSecOps para Inserção e Automação de Práticas de Segurança em Pipelines CI/CD

Autor: Guilherme Henrique de Lima Machado

Orientador: Prof. Lesandro Ponciano

Instituição: PUC Minas

Estrutura da Apresentação

1

Introdução

Apresentação formal da pesquisa e autores.

2

Contexto e Problema

O cenário atual e os desafios de segurança no desenvolvimento ágil.

3

Objetivos e Pergunta de Pesquisa

Direcionamento e questionamento central do estudo.

4

Fundamentação Teórica

As bases conceituais que sustentam a pesquisa.

5

Metodologia Proposta

O desenho experimental e as ferramentas utilizadas.

6

Resultados Esperados

O que se pretende alcançar e validar com o projeto.

7

Status e Próximos Passos

Estado atual e o planejamento futuro da pesquisa.

Contexto e Problema

Contexto: O DevOps revolucionou a entrega de software, acelerando o ciclo de desenvolvimento e implantação.

Problema: A segurança tradicional, com seus processos reativos, falha em acompanhar o ritmo do DevOps, tornando-se um gargalo crítico ou simplesmente ignorada em prol da agilidade.

Consequência: Softwares são frequentemente implantados em produção com vulnerabilidades conhecidas, expondo as organizações a riscos significativos.

Lacuna: Há uma carência de exemplos práticos e claros sobre como orquestrar múltiplas ferramentas de segurança de forma eficiente dentro de um único pipeline CI/CD.

Objetivos e Pergunta de Pesquisa

Pergunta Central

"Como incorporar práticas de segurança de forma contínua e automatizada em pipelines CI/CD?"

Objetivo Geral

Implementar uma abordagem prática para a inserção e automação eficaz de segurança em pipelines CI/CD.

Objetivos Específicos

Integrar práticas de segurança de forma fluida no fluxo de CI/CD.

Aplicar ferramentas automatizadas (SAST, DAST, SCA, IaC) em um ambiente experimental controlado.

Validar a efetividade da detecção de vulnerabilidades em um cenário controlado.

Fundamentação Teórica: Pilares do DevSecOps

DevSecOps

Cultura de responsabilidade compartilhada pela segurança, com foco no princípio de Shift-Left – integrar segurança desde as etapas iniciais do desenvolvimento.

Continuous Security Testing

Implementação de testes de segurança automatizados que são executados a cada alteração de código (*commit*) ou construção (*build*), garantindo feedback rápido sobre vulnerabilidades.

Pipeline CI/CD

A esteira de automação é utilizada como a principal plataforma para governança de segurança, orquestrando as ferramentas e processos de forma integrada e contínua.

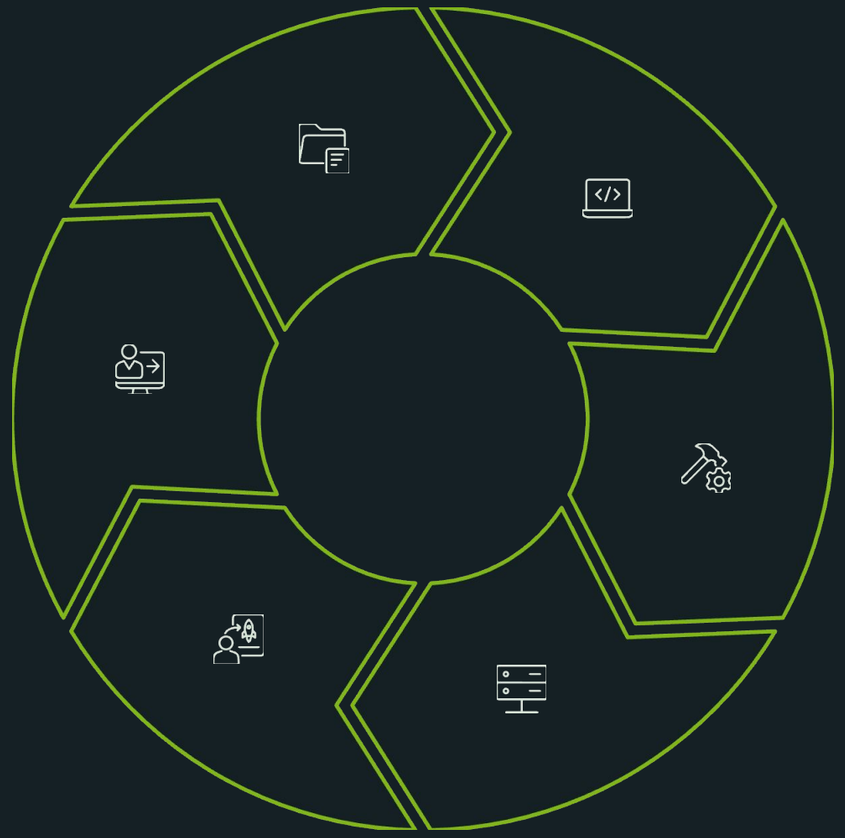
Metodologia Experimental: O Pipeline Proposto

Data

COLETA E ANÁLISE

Code

Semgrep (SAST)



Runtime

OWASP ZAP (DAST)

Build

Dependency-Check (SCA)

Deploy

GKE (Kubernetes)

Infra

Checkov (IaC) + Trivy (Container)

Abordagem: Pesquisa Experimental em Ambiente Controlado.

Infraestrutura: Google Cloud Platform (GCP) provisionada via Terraform (IaC).

Aplicação Alvo: DVWA (Aplicação intencionalmente vulnerável para validação dos testes).

Resultados Esperados e Próximos Passos

Resultados Esperados

1

Viabilidade Técnica

Demonstrar a orquestração de ferramentas open source sem comprometer a agilidade do desenvolvimento.

2

Detecção Efetiva

O pipeline deve detectar vulnerabilidades conhecidas (ex: SQL Injection, XSS) da DVWA e gerar alertas.

3

Artefatos

Geração de relatórios centralizados de vulnerabilidade no Cloud Storage para auditoria.

Status e Próximos Passos

1

Status Atual:

✓ Revisão Bibliográfica.

✓ Definição da Arquitetura e Ferramentas.

✓ Provisionamento da Infraestrutura (Terraform).

2

Próximos Passos:

- Execução completa dos testes com a DVWA.
- Coleta e análise qualitativa dos relatórios gerados.
- Escrita da análise final e conclusão.

Obrigado!

Autor: Guilherme Henrique de Lima Machado - guilhermeoh.machado@gmail.com

Orientador: Prof. Lesandro Ponciano

Instituição: PUC Minas