SHIFT
LEFT
SECURITY

DEVSECOPS

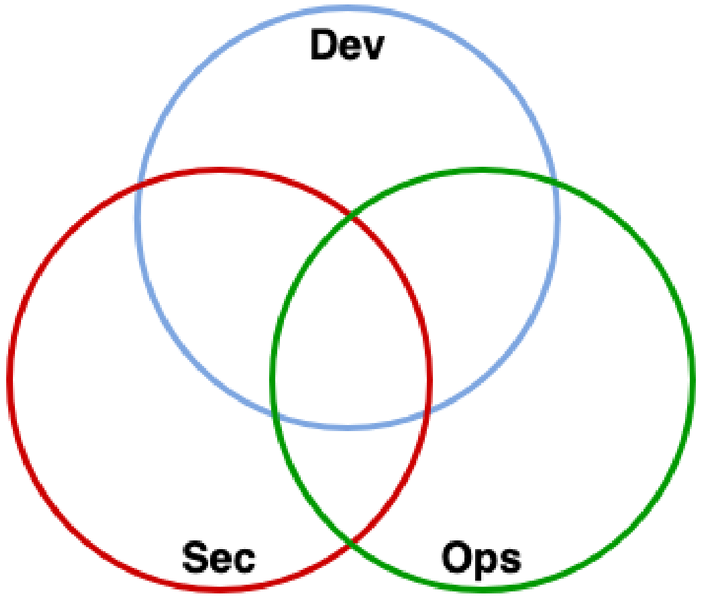
Proaktif Güvenlik Yaklaşımı

Shift Left Security, yazılım geliştirme yaşam döngüsündeki güvenlik testlerinin ve süreçlerinin olabildiğince erken aşamalara kaydırılması yaklaşımıdır.

SDLC Güvenliği Entegre Etmenin Avantajları

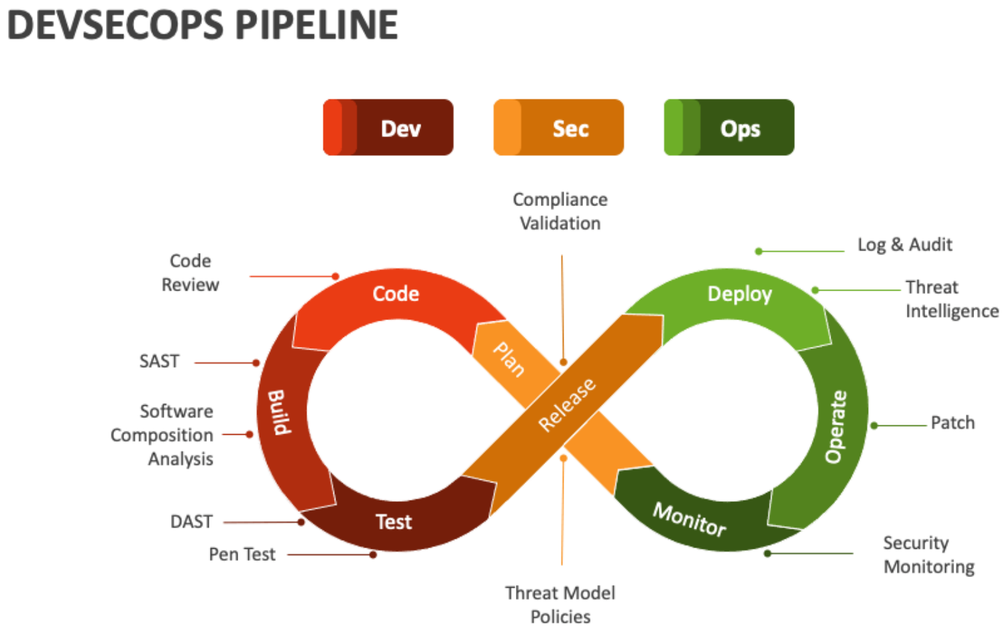
- Yazılım zafiyetlerinin varlığını büyük ölçüde azaltma yeteneği
- Güvenli yazılım geliştirmeye yönelik düzenlemeleri, standartları veya gereksinimleri karşılama yeteneği
- Maliyeti tekrarı azaltma

DevSecOps



Uygulama Güvenliğini Yeniden Tanımlamak

DevSecOps, yazılım geliştirme **(Dev)**, güvenlik **(Sec)** ve operasyon **(Ops)** süreçlerinin birleşimidir. Geleneksel yazılım geliştirme süreçlerinde güvenlik genellikle sona bırakılırken, **DevSecOps** bu yaklaşımı değiştirmiştir. **DevSecOps**, güvenliği yazılım geliştirme ve operasyon süreçlerinin her aşamasına entegre ederek proaktif bir güvenlik kültürü oluşturmayı hedefler.



DevSecOps Aşamaları

01

Planning and Design

02

Coding and Development

03

Testing

04

Integration and Deployment

05

Monitoring and Improvement

06

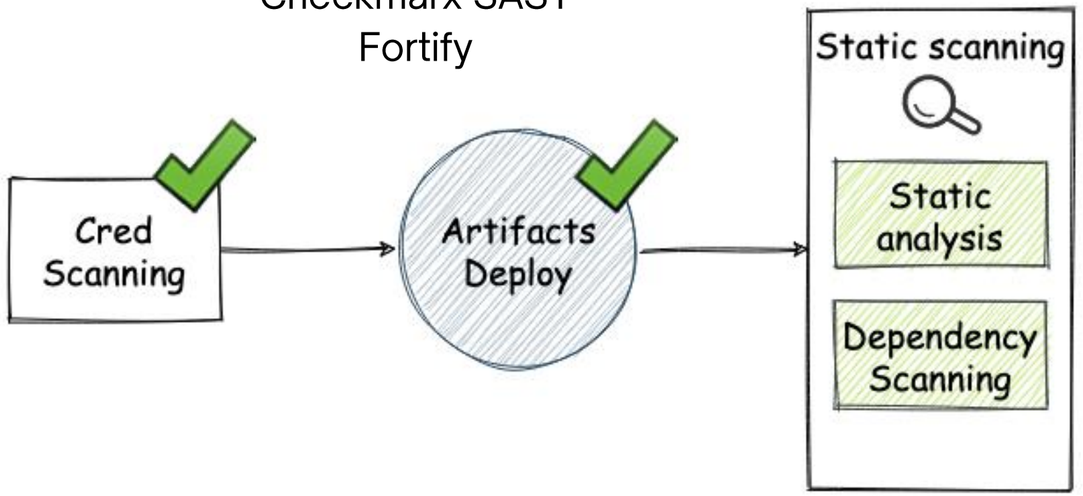
Education

SAST(STATIC APPLICATION SECURITY TESTING)

Statik Kod Analizi genellikle Kod İncelemesinin (**white-box testing**) bir parçasıdır ve programı çalıştırmadan kodu inceleyerek yapılan bir hata ayıklama yöntemidir.

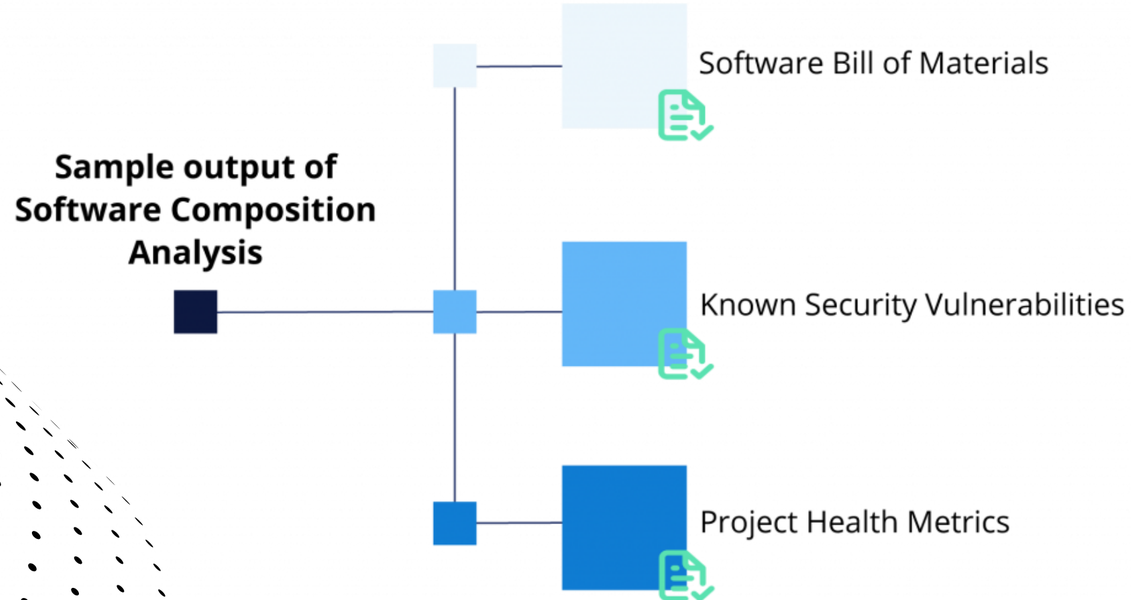
Tools:

SonarQube
Veracode
Security Code Scan
Brakeman
Enlightn
Inquisition
CodeSweep
HCL AppScan on Cloud
Semgrep
Checkmarx SAST
Fortify



Software Component/Composition Analysis (SCA)

Software Component Analysis, üçüncü taraf ve açık kaynak bileşenlerini yönetmek için uygulama güvenliğini otomatikleştirme sürecidir. SCA (**Software Component Analysis**), kod tabanımızda potansiyel güvenlik açıklarına sahip bileşenleri tespit ederek, **Supply-Chain Attack** gibi yüksek güvenlik risklerini önlemeyi sağlar. Ayrıca her bileşenin lisans bilgilerini de sunar. Bu sayede, organizasyonların kod tabanındaki kütüphanelerdeki güvenlik risklerini azaltmalarına yardımcı olur.

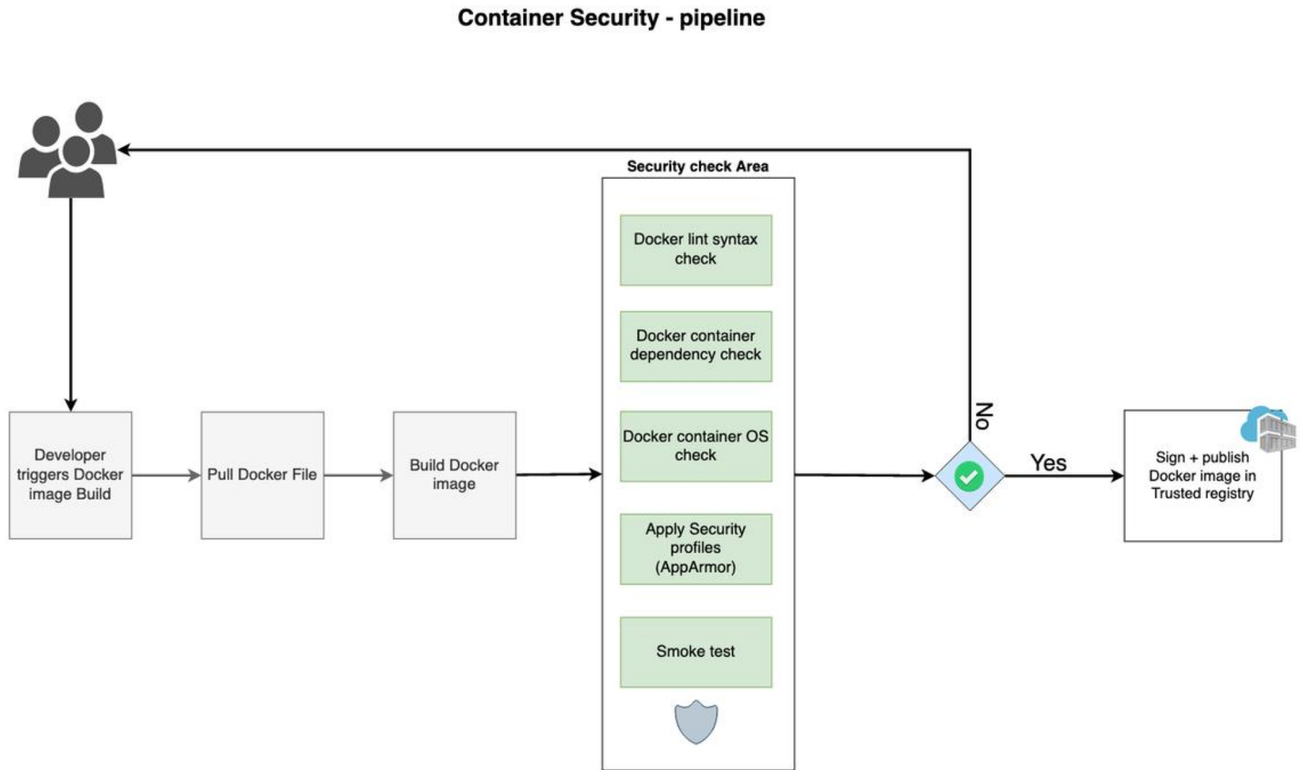


Tools:

OWASP Dependency Check
RetireJS
Safety
bundler-audit
Hakiri
HCL AppScan on Cloud
Snyk

Container Vulnerability Scanning

Container, yazılım geliştirme ve deployment süreçlerinde popüler hale gelmiştir. Ancak container, birçok component ve layer'dan oluşur, bu da potansiyel güvenlik risklerini beraberinde getirir. Container güvenliği, imajların ve bunların çalıştırıldığı ortamların güvenliğini sağlamak amacıyla yapılır.



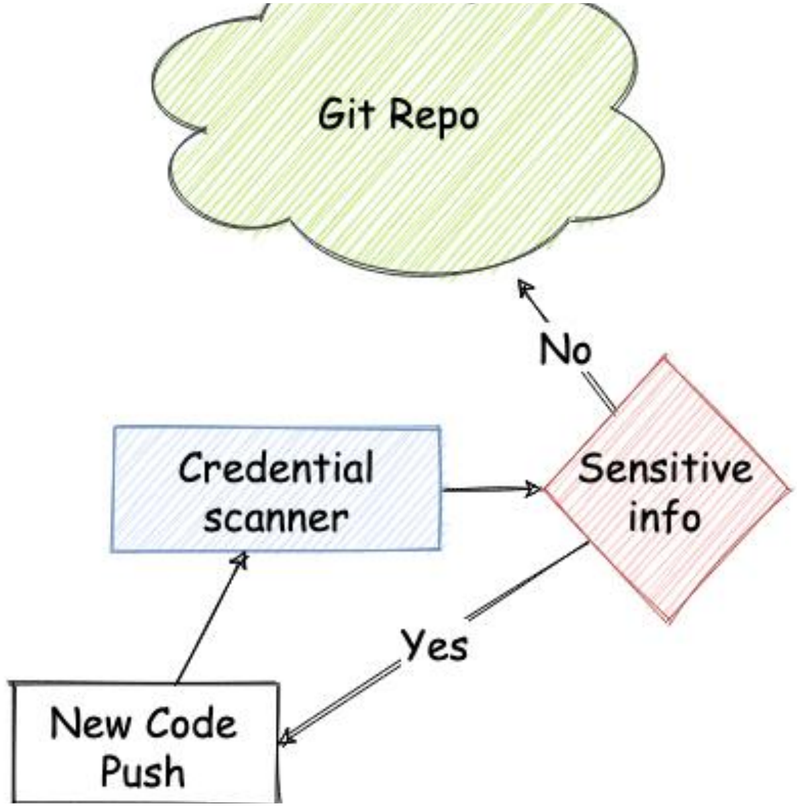
Tools:
Clair
Anchore
Trivy

Secrets Management

Secret Management, yazılım geliştirme yaşam döngüsünde güvenliğin sağlanması açısından kritik bir süreçtir. Bu süreçte, şifreler, API anahtarları, gibi hassas bilgilerin güvenli bir şekilde yönetilmesi hedeflenir. HashiCorp Vault, AWS Secrets Manager ve Azure Key Vault gibi araçlar, gizli bilgilerin güvenli bir şekilde saklanmasını ve erişimini sağlar.

Tools:

gittyleaks
git-secrets
Repo-supervisor
truffleHog
Git Hound



Infrastructure as Code (IaC) Scanning

Infrastructure as Code (IaC) Scannig yazılım güvenliğinin temel taşlarından biridir. Kod tabanlı yapılandırmalar, yanlış yapılandırmalar, güvenlik açıkları taşıyabilir. Bu risklerin erkenden tespit edilip düzeltilmesi, potansiyel güvenlik ihlallerini önlemeye yardımcı olur. **DevSecOps** prensipleri doğrultusunda, güvenlik her aşamada dikkate alınmalıdır ve IaC'nin güvenliği de bu yaklaşımın bir parçasıdır.

Tools:

Checkov
ansible-lint
puppet-lint
tfsec
terrascan
tflint

Dynamic Application Security Testing (DAST)

DAST, “BlackBox” testi olarak bilinir ve çalışan bir uygulamadaki güvenlik açıklarını için zararlı payload enjekte eder. Bu sayede SQL enjeksiyonları veya (XSS) gibi saldırılara yol açabilecek potansiyel kusurları tespit eder.

Tools:

Acunetix
Netsparker
InsightAppSec (AppSpider)
Veracode Dynamic Analysis
Burp Suite
HCL AppScan on Cloud
Nuclei

Teşekkürler

Emre Can VURAL
Application Security at Trendyol Group


info@emrecanvural.com


emrecanvural

emrecanvural.com