Incorporating Security in DevOps Process

Chintan Gurjar

Traditional DevOps (Agile) Security Challenges

Challenge	Recommendation
Change management process slower your delivery time	Change management should be automated whenever it is feasible
Security is not embedded in SDLC process	Integrate security in each and every phase of the SDLC and DevOps
Threat analysis and modelling is not incorporated in agile development	A dedicated security team should perform threat analysis and modelling during each phase of agile
Compliance is handled post deployment	Compliance controls are simultaneously documented, defined and audited during the development
DevSecOps team lacks in security knowledge	Security trainings should be given to project managers, developers and network engineers who are part of agile development process

DevOps 101

Development ←→ Operations

Integrate
Collab
Communicate

Integrate: Use tools, methods to integrate development workflow and become more proactive and reactive

Collaborate: All phases of Agile process for development and operations work together

Communicate: Seamless communication for decision making process, solving hurdles, increasing visibility and transparency

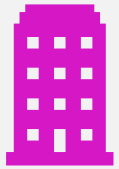
Traditional method of security

Security team defines security requirements in development phase

Developers pause or don’t start development phase until all security requirements are met

This slows entire development process

Ratio of Developers vs Security professionals



On an average there is only 1 security person against 20 developers in any company

Considering this ratio it is hard for security person to take responsibilities of entire product development

Developers need to be educated in application security and secure coding practices

Determining need of DevSecOps

- **Malicious user's mindset –** Open conversation with developers and ask them to act as evil users. Come up with stories and ideas how an attacker can abuse xyz application functionality. These stories and ideas would lead team towards the security requirement.

Determining need of DevSecOps cont.

- **Expert's advise –** Security persons would know better impact and challenges in product development. Make one of the security person the development team champion who would educate developers about the potential challenges of product development from security point of view.

Determining need of DevSecOps cont.

- **Threat Modelling –** Implement and execute STRIDE threat modelling which will enable organisation to define security requirements for the product development.

DevSecOps learning requirements

- **OWASP training –** Train developers on top 10 application security threats and this training should include:
  - What are these threats?
  - Why they occur?
  - What type of vulnerable code is responsible for these threats?
  - What are different ways to fix this on application and server level?

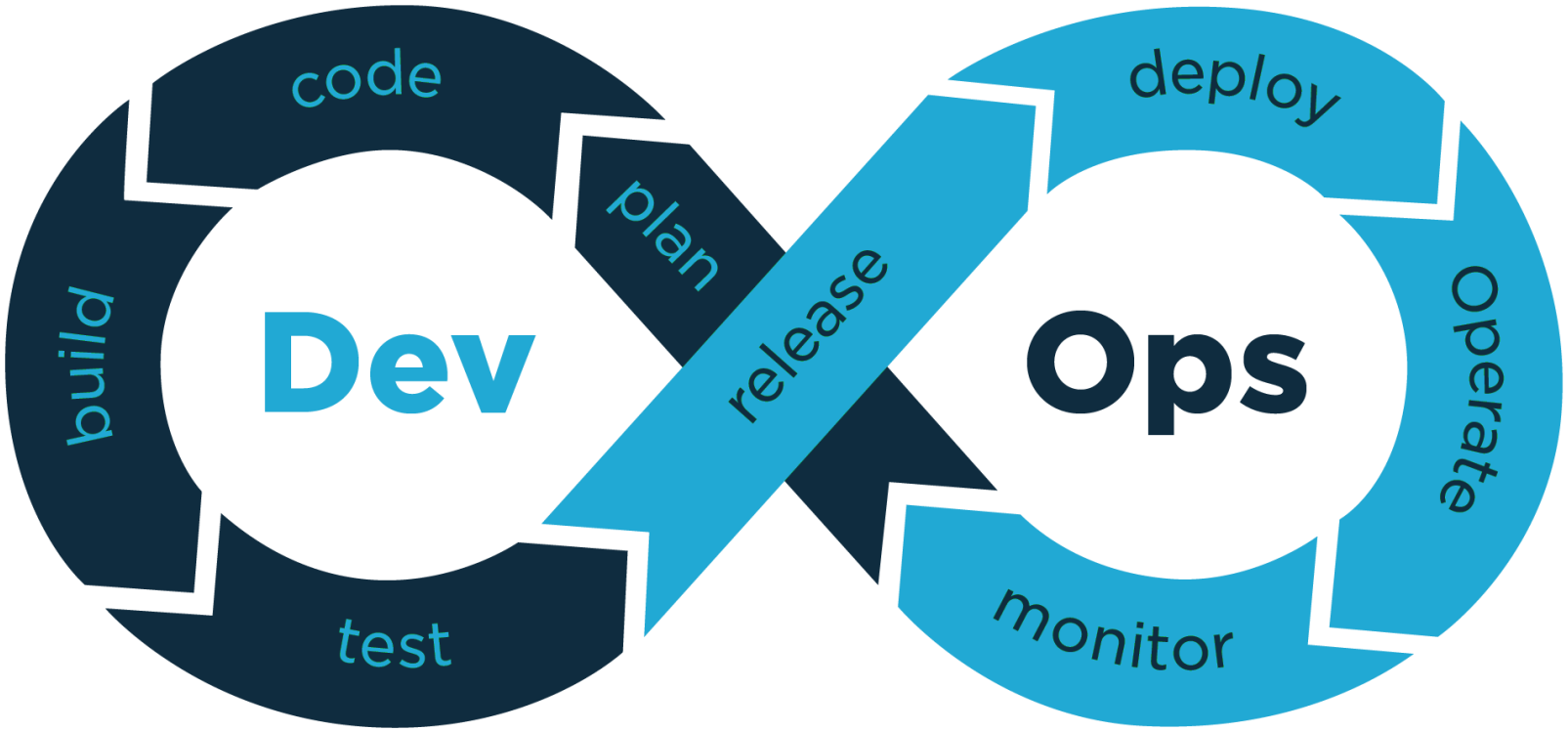
DevSecOps learning requirements cont.

- **Secure coding practices -** Train developers on secure coding practices using external vendors who are expert in providing such corporate trainings. If budget is the constraint then train your developers using freely available OWASP material located below:

  https://www.owasp.org/index.php/OWASP_Secure_Coding_Practices_-_Quick_Reference_Guide

  https://www.owasp.org/index.php/OWASP_Secure_Coding_Practices_Checklist#General_Coding_Practices

DevOps model



Automation is the key

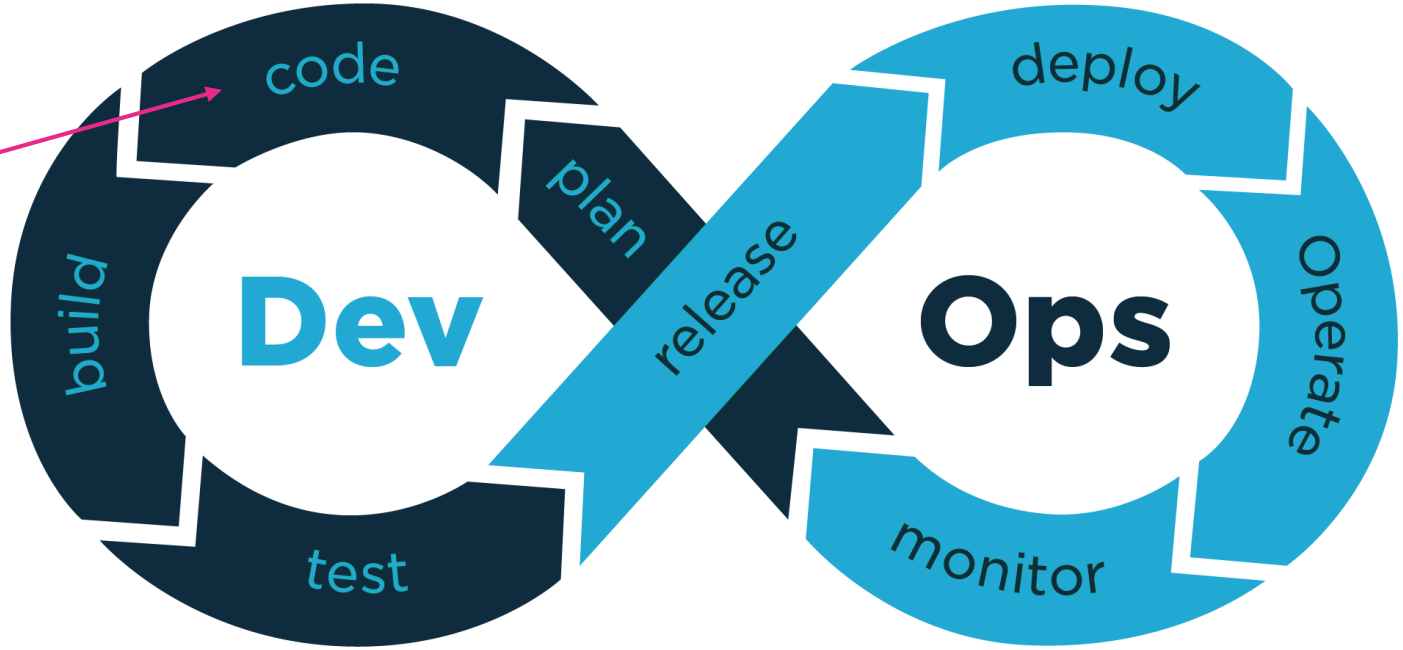
In following slides you will see all the tools and solutions that can be integrated with the DevOps model in order to automate security part.

Automation accelerates agile development process.

DevSecOps automation

Static Application Security Testing (SAST) identifies vulnerabilities in the source code:

- SonarQube
- CxSAST (Checkmarx)
- Fortify (Micro Focus, Formally HP)
- Veracode Static Analysis (Veracode)

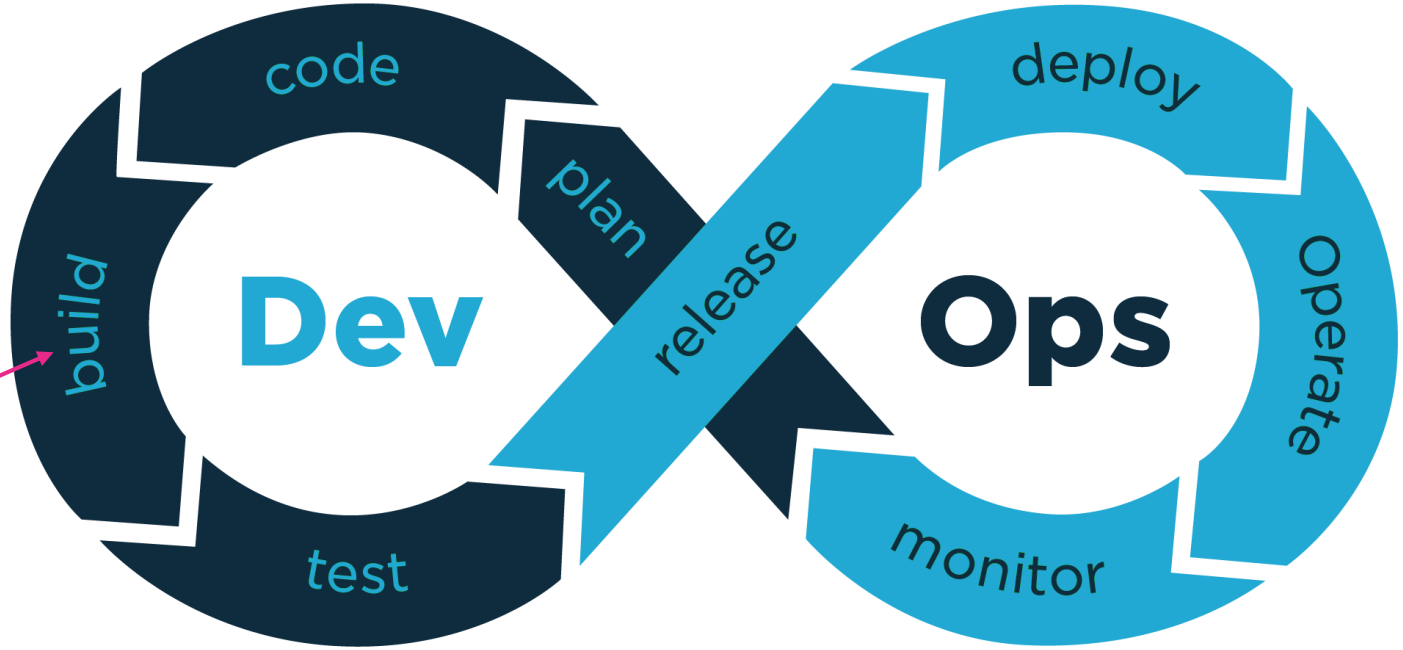


https://www.owasp.org/index.php/Source_Code_Analysis_Tools

DevSecOps automation cont.

Software composition analysis (SCA) enables developers to analyze and manage the open-source elements of their applications such as libraries

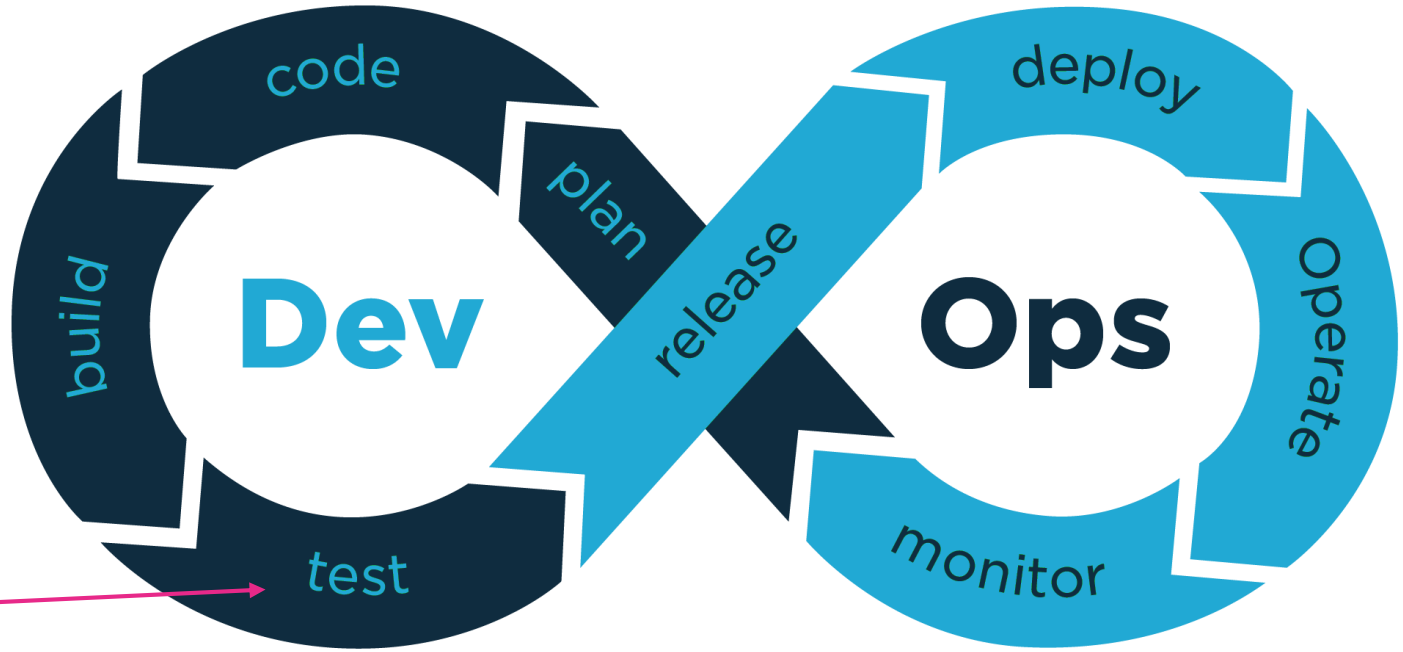
- Blackduck
- WhiteSource
- Snyk
- Threatwatch
- CAST Highlight
- Dependency-Track
- Veracode Software Composition Analysis
- Whitehat Sentinel SCA Essentials



DevSecOps automation cont.

Dynamic application security testing (DAST) identifies vulnerabilities in running application:

- Acunetix WVS
- IBM Appscan
- Netsparker
- Burp Suite

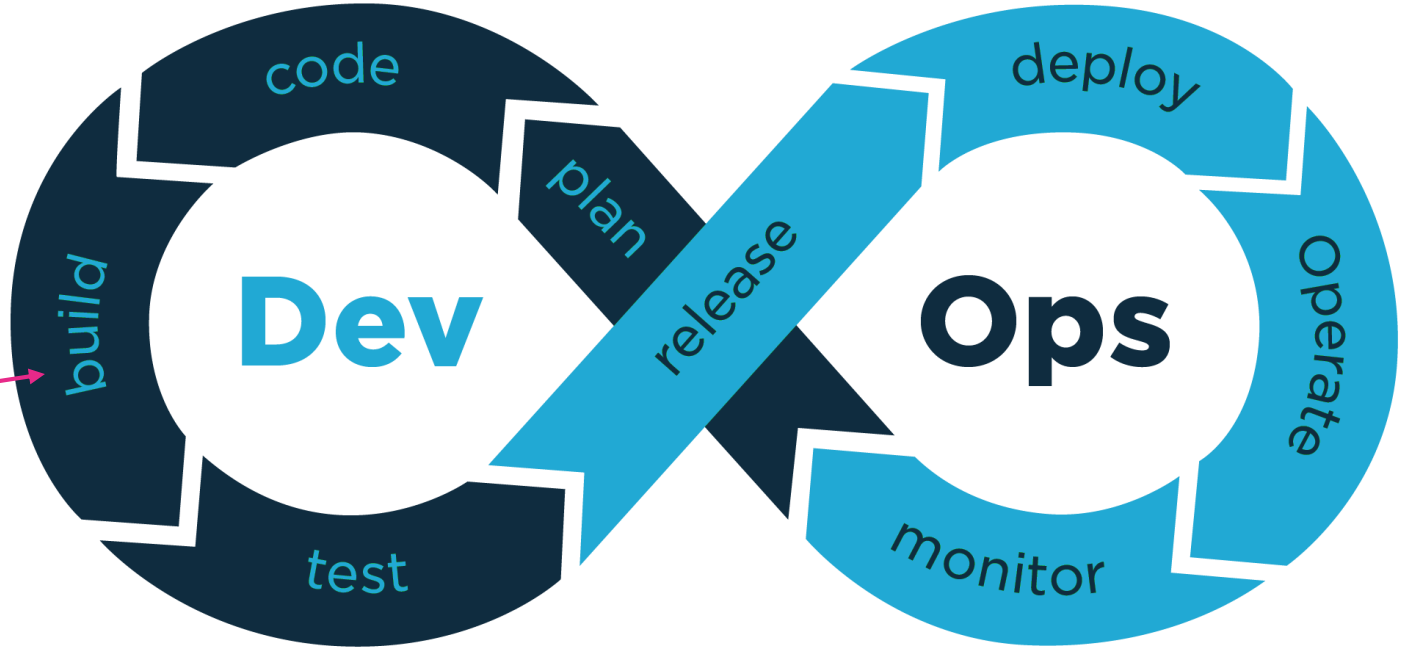


https://www.owasp.org/index.php/Category:Vulnerability_Scanning_Tools

DevSecOps automation cont.

Container security software are used to improve the security configurations of the containers:

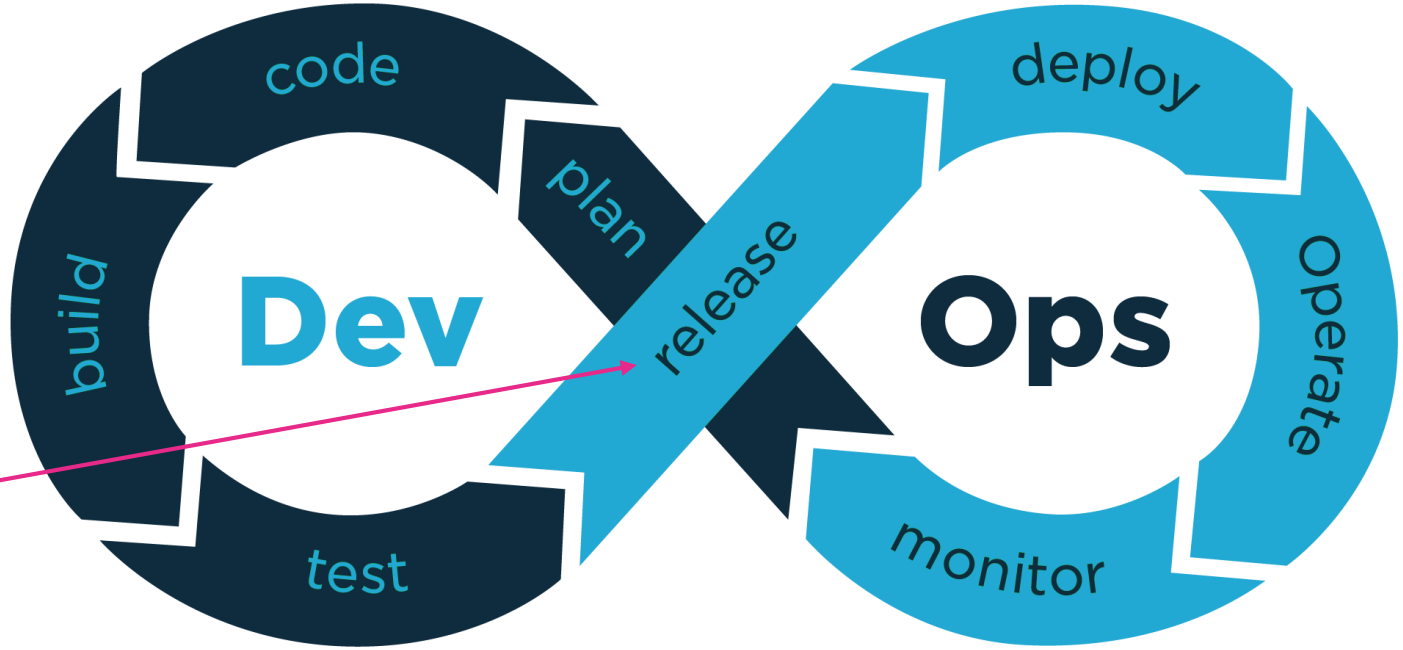
- Clair
- Anchore Enterprise
- Portshift
- WhiteSource for Containers
- Alcide
- Aqua Cloud Native Security Platform
- Deepfence Security as a Microservice
- NeuVector Kubernetes Container Security Platform
- Qualys Container Security
- StackRox Kubernetes Security Platform
- Sysdig Secure



DevSecOps automation cont.

Vulnerability scanning activity to find vulnerabilities in released application/software.

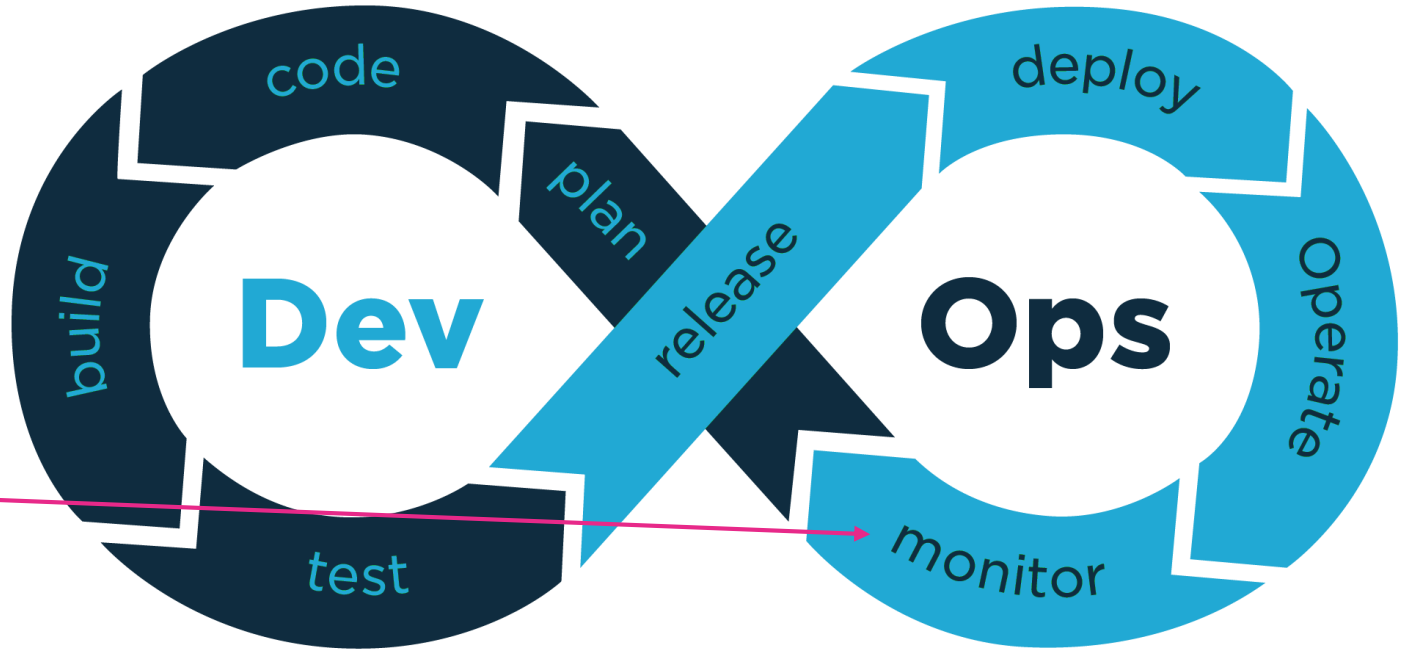
- OWASP ZAP
- Metasploit
- Nexpose
- Nessus
- Core Impact
- Qualysguard



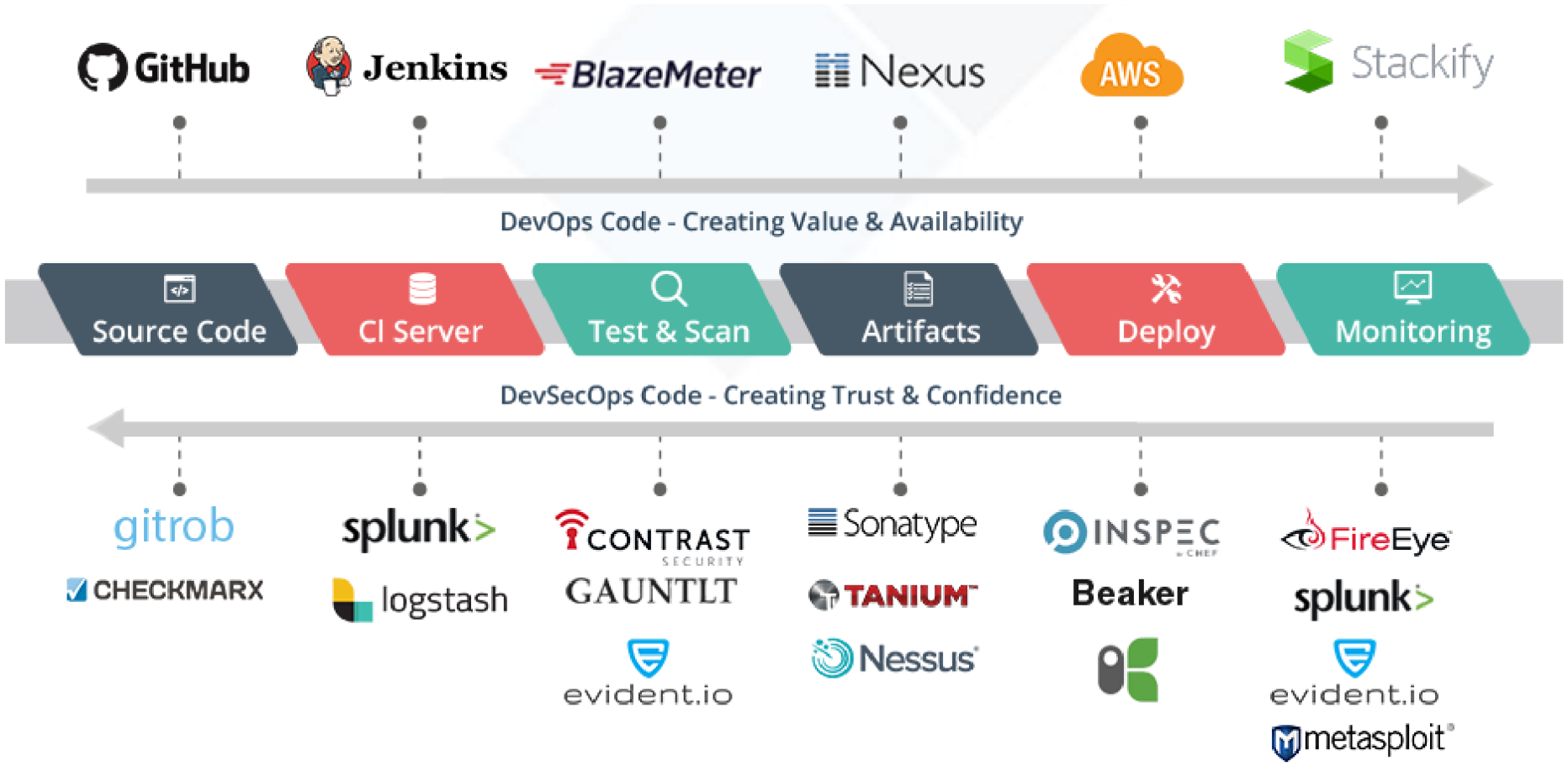
DevSecOps automation cont.

Fuzzing activity to find vulnerabilities due to improper random data.

- Burp Suite
- Peach Fuzzing Platform
- Spirent Avalanche NEXT
- Beyond Security's beSTORM product
- Codenomicon's product suite



Tools integration/use for each phase of DevSecOps



How to measure effectiveness of your DevOps security

- **Number of vulnerabilities identified in code repository**
  - How to find – Automated code security audit tools (Open source and commercial)
  - Which activity is associated with this – Code development

How to measure effectiveness of your DevOps security cont.

- **Number of vulnerabilities identified in application build**
  - How to find – Vulnerability scanners (Open source and commercial)
  - Which activity is associated with this – Application deployment

How to measure effectiveness of your DevOps security cont.

- **Number of vulnerabilities identified in application build**
  - How to find – Penetration testing activity (Internal security team and external vendor)
  - Which activity is associated with this – Application deployment

How to measure effectiveness of your DevOps security cont.

- **Number of false positive ratio identified in entire DevOps process**
  - How to find – Evaluate all vulnerabilities identified by automated scanners, code repository scanners, penetration activity results and measure the ratio of legitimate vulnerabilities vs. false positive rate
  - Which activity is associated with this – Application deployment

How to measure effectiveness of your DevOps security cont.

- **Number of number of attacks/threats post deployment process**
  - How to find – SIEM/WAF/IDS/IPS logs are evaluated by SOC team to find legit attacks against their application
  - Which activity is associated with this – Post deployment monitoring activity

Once you have all numbers, you can identify whether your security incorporation in the DevOps process is effective or there are areas of improvement

Security by People Process Technology

People

Build security culture

Provide vendor training

Provide cross team training

Awareness

Process

Security in orchestration

Formal security process in updates and upgrades

Version control and metadata

Compliance

Threat intelligence program management

SIEM

Approvals and peer review pre-development

Incident management

Red-team, blue-team practices

Bug bounty program management

Technology

Configuration management

Code security

Build security

Host hardening

Patch management

Auditing and scanning

Binary scanning

Pre-deployment auditing, testing and scanning

Managing secret keys, code, passwords of application

Security Controls by DevOps model

Governance

Strategy and metrics

Policy and compliance

Education and guidance

Design

Threat assessment

Security requirement

Security architecture

Implementation

Secure build

Secure deployment

Defect management

Verification

Architecture assessment

Requirement driven testing

Security testing

Operations

Incident management

Environment management

Operational management

DevSecOps checklist

- Implement RBAC and IAM solutions
- Implement threat modelling techniques
- Classify data from security point of view
- Implement risk management and assessment processes and procedures
- Train developers on secure coding
- Segregate production and UAT environment along with data being used in both of them
- Use SAST and DAST tools to find security issues in application code and final build
- Test security for third party libraries and components via penetration testing and vulnerability scanning
- Test security for containers and databases via penetration testing and vulnerability scanning
- Comply with industry standard frameworks
- Restrict access to production environment
- Perform peer code review
- Perform unit testing and functional security testing

DevSecOps checklist cont.

- Automate security processes on coding and monitoring phase
- Apart from apps, monitor security for entire infrastructures
- Implement tools that gives you actionable insights of security issues
- Provide security to physical infrastructure
- Isolate and segregate dockers and Kubernetes
- Test, secure and harden cloud environment
- Have an effective assessment management processes for app components
- Secure API and database
- Secure data at rest and data in motion
- Provide security awareness trainings to your employees
- Use cutting edge NGFW/IDS/IPS tools to protect production environment
- Perform external and internal shadow IT activity to identify hanging fruits
- Implement periodic security management and review practices in DevOps environment

Benefits of DevSecOps

References

- https://medium.com/tech-tajawal/devops-in-a-scaling-environment-9d5416ecb928
- https://www.opcito.com
- https://owaspsamm.org/model/