

DevSecOps - Background, Status and Future Challenges

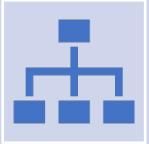
F. Ozgur Catak

University of Stavanger

<https://github.com/ocatak/devsecops-tutorial>



DevOps



A philosophy, aims at building up a culture of collaboration between originally isolated teams.

Departments

- Software development
- deployment operations



Improving the efficiency by eliminating the boundaries between these two phases of development.

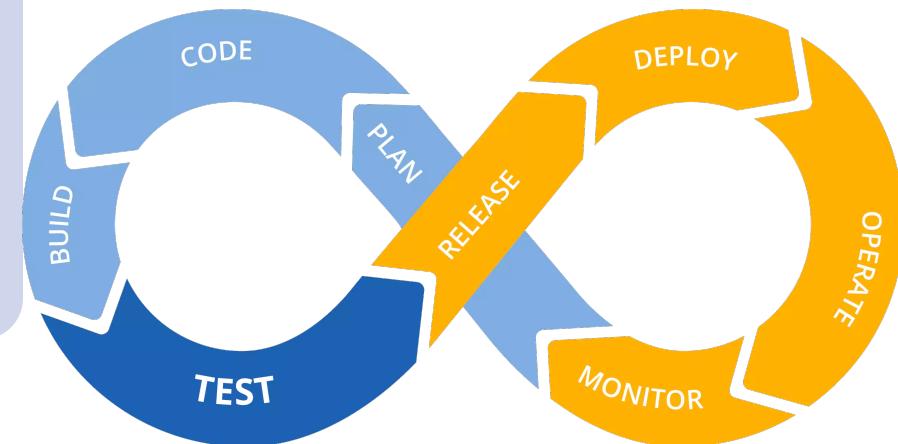
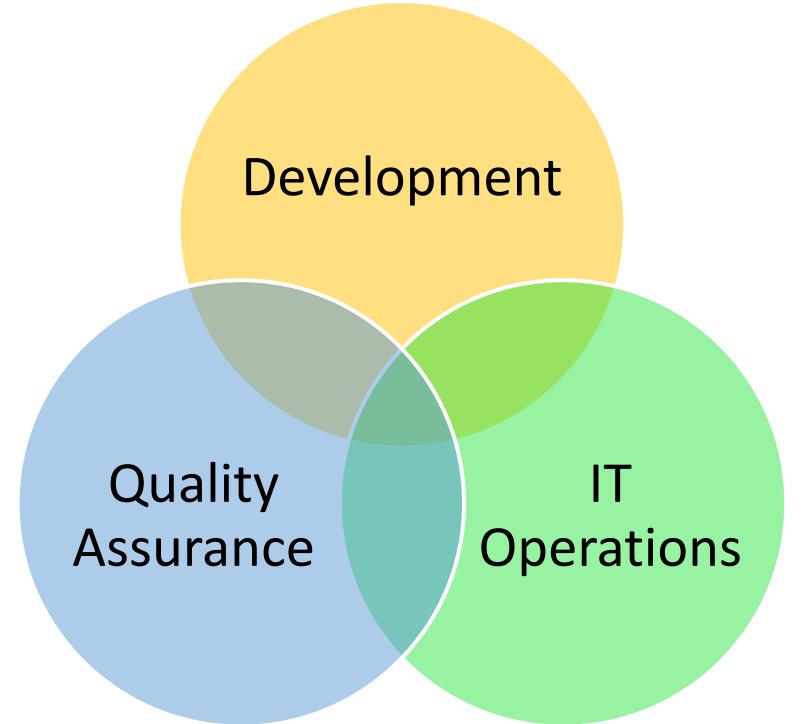


Includes

Continuous integration, where the coding, building, integration, and testing processes are carried out.

Continuous delivery, which includes continuous integration but mainly focuses on product delivery.

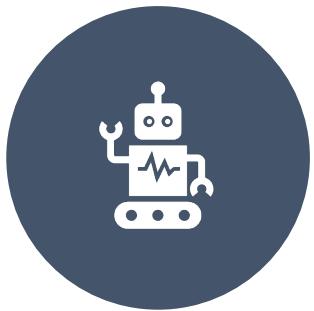
Continuous deployment, which aims at automating project deliveries.



Defining DevOps: C-A-M-S



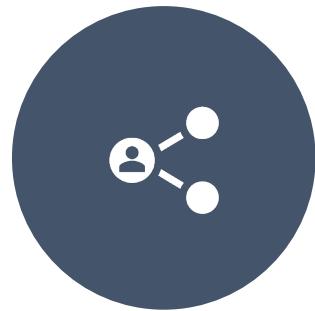
CULTURE



AUTOMATION



MEASUREMENT



SHARING

Culture

- Primary characteristics of DevOps culture: Increased collaborations
- Dev-Ops interact for a single target: to break down the barries



INTERACTION



TARGET



TEAMS

DEVELOPMENT - OPERATION

Automation

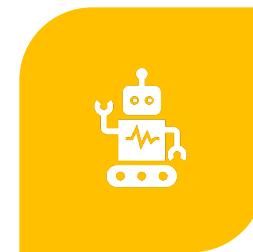
- Keyword: automation
- Speed-up feedback between teams
- The app will be delivered to the customers easily and successfully.
- DevOps pipeline: flow of information
- New functionality in the app.
 - New library, New attack surface, SQLi?



FEEDBACK



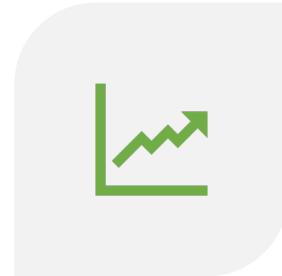
PROCESS



TOOLS FOR
COLLABORATION

Measurement

- Track improvement
- See bottlenecks/problems easily
- Performance of the production system
 - Production system didn't degraded anything like response time,



PERFORMANCE



MEASUREMENT



BOTTLENECKS

Sharing

- Sharing ideas, problems, and solutions between various teams
- By transferring the knowledge, the collective intelligence increases and benefits for everyone.



IDEA



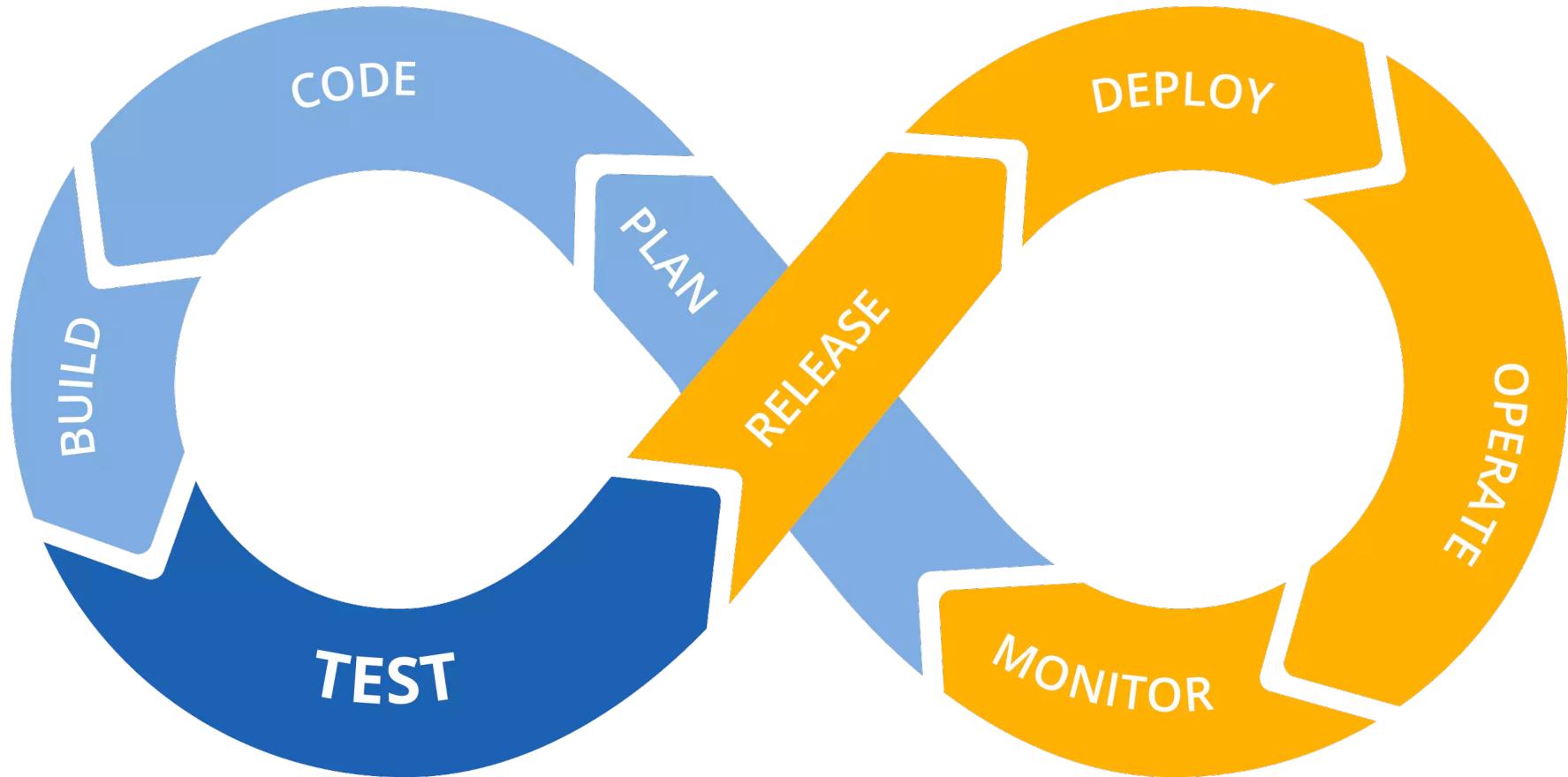
SHARING



TALK

DevOps Lifecycle

- 8 stages in the
 - Planing
 - Development
 - Operation



DevOps Tools and Technologies

Version control systems: Git, SVN, Mercurial, etc.

Continuous integration (CI): Jenkins and TeamCity

Containers: Docker or Vagrant.

Framework automation tools: Python, Shell, or Bash.

Cloud services: Microsoft Azure, Google Cloud, Amazon Web Services

Traditional Security Perspective: Penetration testing

WHAT IS A PENETRATION TEST?



An authorized attack on a computer system, network, or application to identify security vulnerabilities bad actors might exploit.

The Process



Types of Penetration Tests

Application		Network
Physical		IoT/Mobile

Why Conduct a Penetration Test?



Identify
security
vulnerabilities



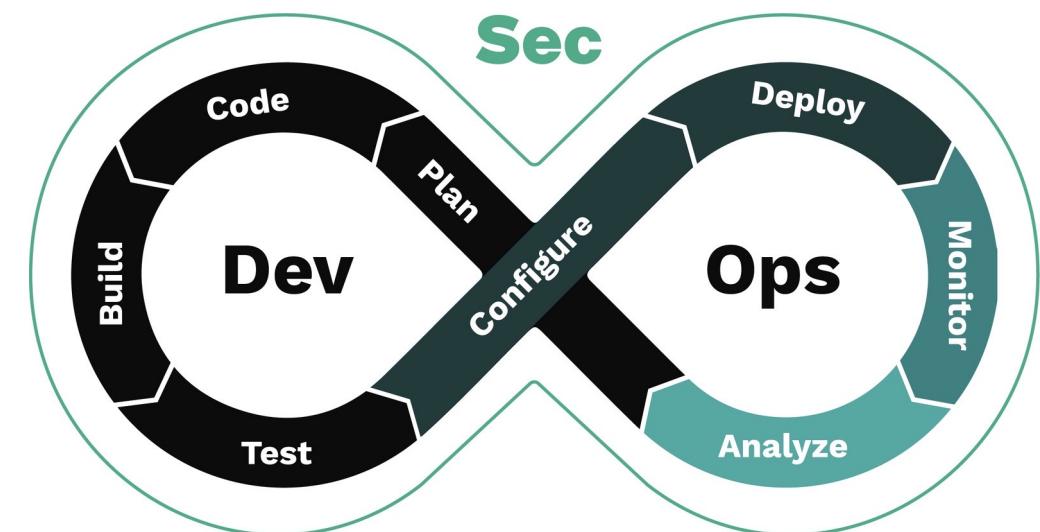
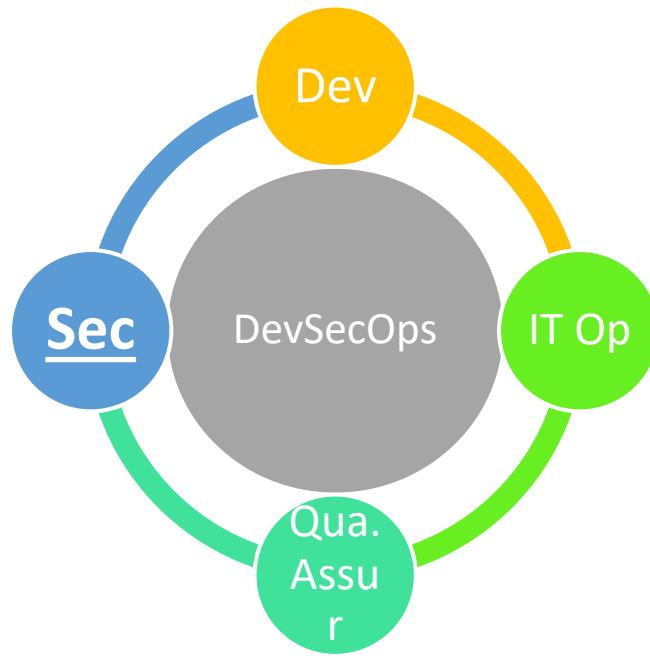
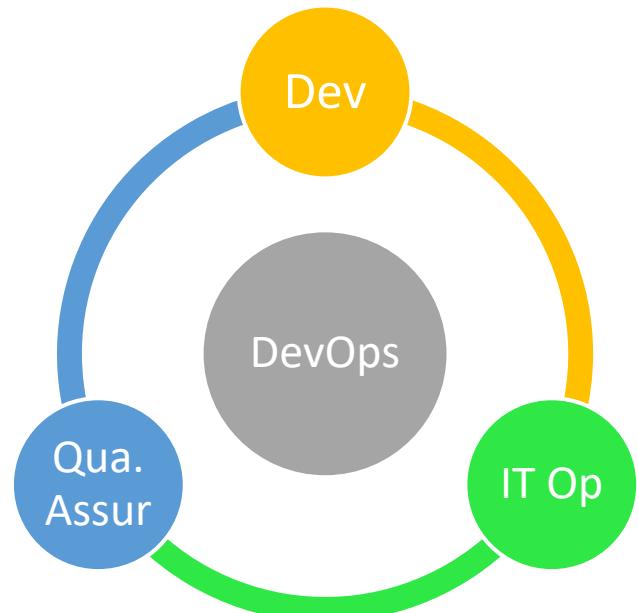
Validate
compliance with
policies



Evaluate
effectiveness of
defenses

DevSecOps: what's and differences from DevOps?

- It is a philosophy of **integrating security methods** into a DevOps process.
- From the very start of a **SDLC**, DevSecOps works to make the application secure by introducing a variety of security techniques.
- DevOps heavily relies on automation. The same is true for DevSecOps, which aims at automating every aspect, including security audit.



Why DevSecOps

Companies implementing DevSecOps address vulnerabilities faster than others

Higher speed and agility: security teams

Quality and compliance assurance

Implement, Educate & Follow best practices for secure coding and infrastructure provisioning/ deployment / configuration.

DevSecOps: Securing CI/CD

Integrating security in the CI/CD pipeline which helps minimizing threats/vulnerabilities with every code check in.

Automation of security part.

- Minimize the manual Pentest effort.

To bridge the gap between Dev. Sec and Ops teams

DevSecOps: Upcoming hot buzz

As DevOps & agile enables speed, it comes with risk as well.

Frequent code changes, expanding attack surfaces, day by day growing number of hacks is building the market for DevSecOps

80% of threats & vulnerabilities can be tackled if we enable DevSecOps (20% effort). Famous 80/20 rules or “Pareto principle”

Cloud Security Alliance: The Six Pillars of DevSecOps: Automation

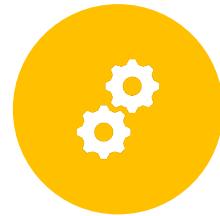
The **Cloud Security Alliance** (CSA) is the world's leading organization dedicated to defining and raising awareness of best practices to help ensure a **secure cloud** computing environment.



COLLECTIVE
RESPONSIBILITY



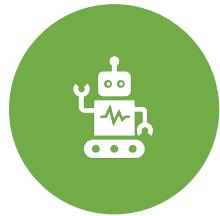
COLLABORATION AND
INTEGRATION



PRAGMATIC
IMPLEMENTATION



BRIDGING
COMPLIANCE AND
DEVELOPMENT



AUTOMATION



MEASURE, MONITOR,
REPORT AND ACTION

Why do we need DevSecOps

- implement and maintain the compliance requirements
- minimize security bottlenecks also automate the security checks
- collaborate with Information security team in the upcoming future by 2021

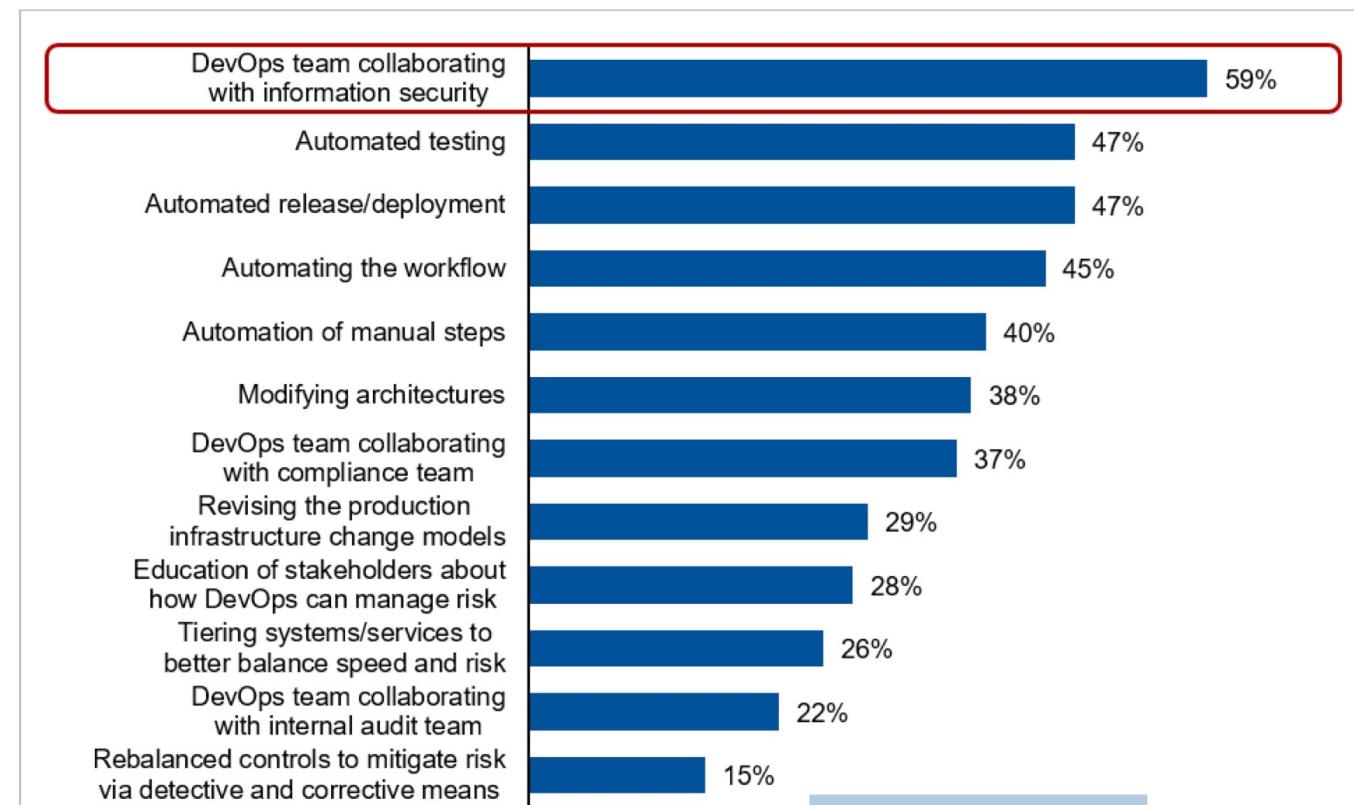


MAINTAIN COMPLIANCE /
REGULATORY REQUIREMENT

SECURITY AUTOMATION /
MINIMIZE SECURITY
BOTTLENECKS

Gartner Report: 10 Things to Get Right for Successful DevSecOps

Figure 1. Strategies to Overcome Hurdles to Using DevOps in Regulated Situations



Practical Transitioning from DevOps to DevSecOps

Static Application Security Testing (SAST)

- White-box security testing
 - Access to the underlying framework, design, and implementation
 - The app is tested from the inside out
 - It is **developer approach**
- Finds vulnerabilities earlier in the SDLC
- Less expensive to fix vulnerabilities
- Can't discover run-time and environment related issues (like configuration vulnerabilities)

Dynamic Application Security Testing (DAST)

- Black-box security testing
 - The tester doesn't know the technologies or frameworks of the app
 - The app is tested from the outside in
 - It is **hacker approach**
- Finds vulnerabilities toward the end of the SDLC
- More expensive to fix vulnerabilities
- Can discover run-time and environment related issues (like configuration vulnerabilities)

DevOps vs DevSecOps

- Traditional Build Phases



- Future Build Phases

devsecops-tutorial

DevSecOps best practices

[View the Project on GitHub](#)
ocatak/devsecops-tutorial

DevSecOps Practical Implementation

<https://github.com/ocatak/devsecops-tutorial>

<http://www.ozgurcatak.org/devsecops-tutorial/>

This project is maintained by [ocatak](#)

DevSecOps - Background, Status and Future Challenges

- F. Ozgur Catak - f.ozgur.catak@gmail.com
- GitHub repository for the lecture at **University of Stavanger**
- June 14th, 2021

Jenkins Installation

we can run Jenkins as Docker image

```
docker pull jenkins
```

Contributing

Pull requests are welcome. For major changes, please open an issue first to discuss what you would like to change.

Please make sure to update tests as appropriate.

License

MIT

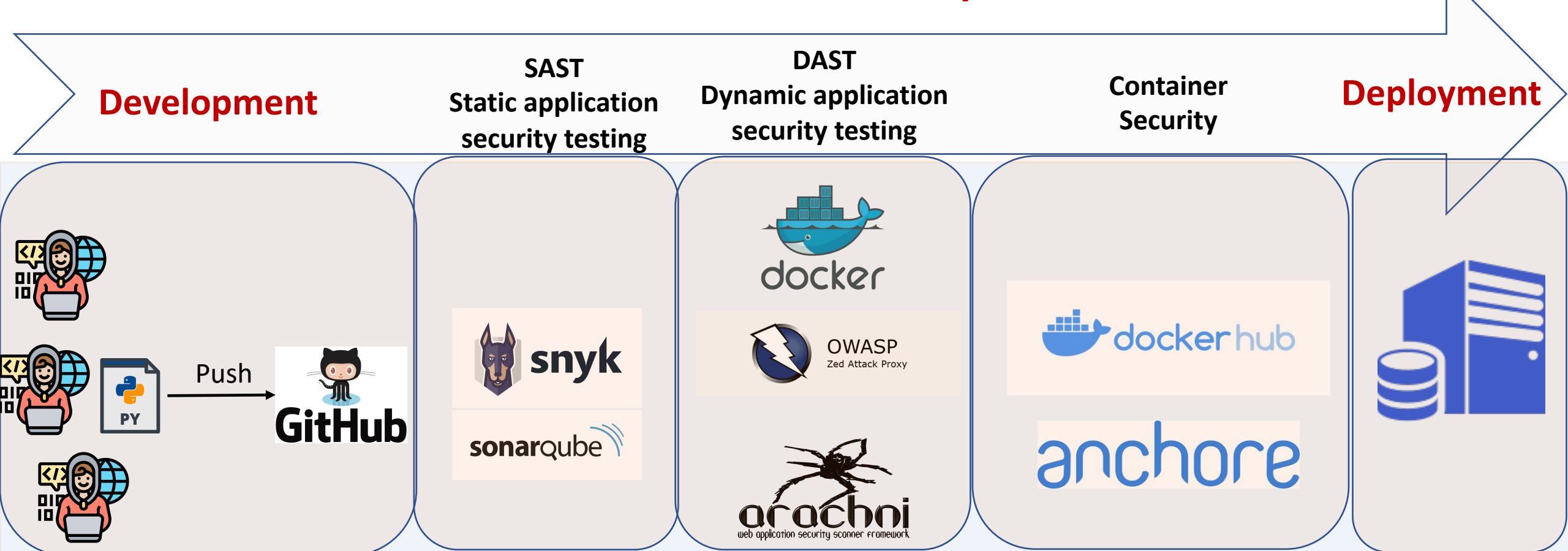
Example Web Application: DevSecOps

<https://github.com/ocatak/devsecops-tutorial>

GitHub repository for the lecture at University of Stavanger



Jenkins Pipeline



Jenkins: Pipeline Creation

Enter an item name

» Required field

Freestyle project

This is the central feature of Jenkins. Jenkins will build your project even used for something other than software build.

Pipeline

Orchestrates long-running activities that can span multiple builds (workflows) and/or organizing complex activities that do not easily fit into a single build step.

Pipeline

Definition

Pipeline script from SCM

SCM

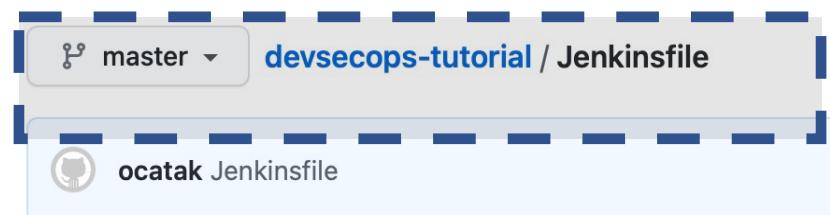
Git

Repositories

Repository URL

```
https://github.com/ocatak/devsecops-tutorial.git
```

Credentials



21 lines (20 sloc) | 267 Bytes

```
1 pipeline{  
2   agent any  
3   stages {  
4     stage ('Build') {  
5       steps {  
6         echo "Build Phase"  
7       }  
8     }  
9     stage ('Test') {  
10       steps {  
11         echo "Test Phase"  
12       }  
13     }  
14     stage ('Deploy') {  
15       steps {  
16         echo "Deploy Phase"  
17       }  
18     }  
19   }  
20 }
```

Declarative:	Checkout SCM	Build	Test	Deploy
	568ms	57ms	34ms	50ms
	568ms	57ms	34ms	50ms

Source Code Security (GitHub)

- Browsing security vulnerability in the *GitHub Advisory DB*
 - <https://github.com/advisories> : It allows you to browse or search for vulnerabilities which affect open source projects on GitHub
 - Below sources are used for vulnerabilities by GitHub
 - The National Vulnerability Database
 - A combination of machine learning and human review to detect vulnerabilities in public commits on GitHub
 - Security advisories on GitHub
 - FriendsOfPHP
- Security alerts for vulnerable dependencies
- Automated security updates
- Update vulnerable dependencies in repository
- Managing alerts for vulnerable dependencies in your organization

The screenshot shows a GitHub repository page for 'ocatak/devsecops-tutorial'. The top navigation bar includes links for Code, Issues, Pull requests, Actions, Projects, Wiki, Unwatch (1), Unstar (2), Fork (0), and a three-dot menu. On the left, a sidebar menu lists Overview, Security policy, Security advisories (0), Dependabot alerts (2, highlighted with a red border), and Code scanning alerts. The main content area is titled 'Dependabot alerts' with a 'Dismiss all' button. It displays two open alerts: one for 'urllib3' with a 'high severity' badge and another for 'requests' with a 'moderate severity' badge. Both alerts are attributed to GitHub and point to requirements.txt files.

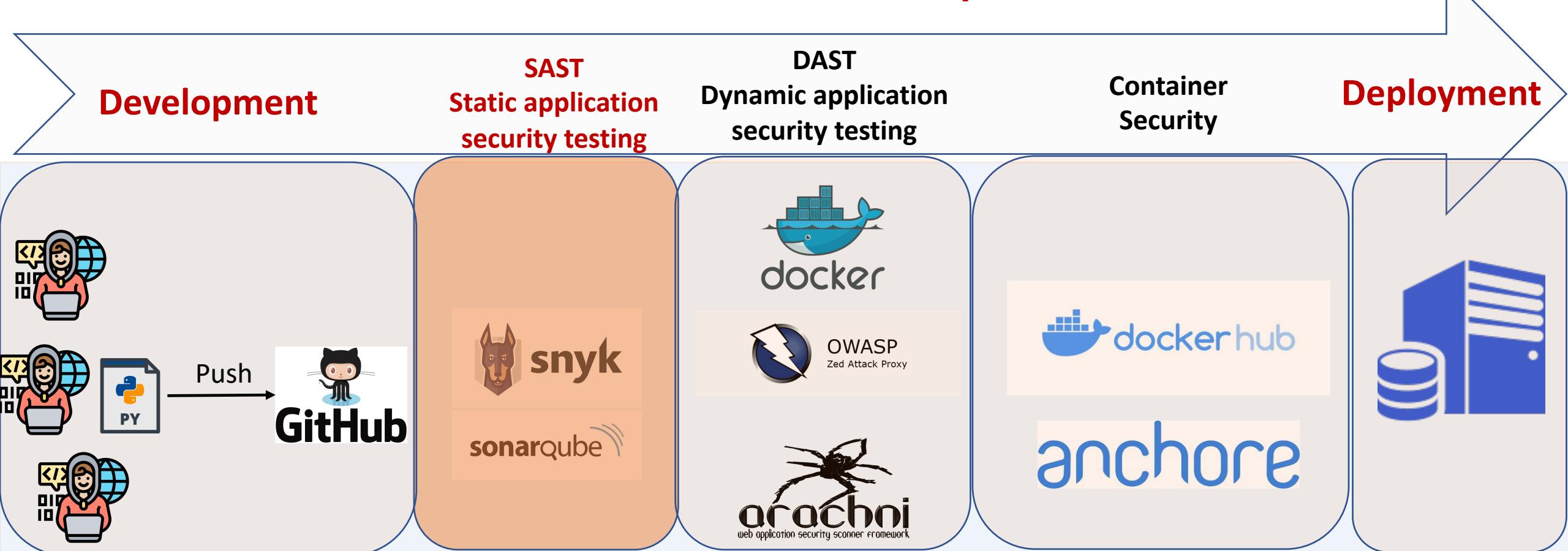
Example Web Application: DevSecOps

<https://github.com/ocatak/devsecops-tutorial>

GitHub repository for the lecture at University of Stavanger



Jenkins Pipeline



DevSecOps: SAST - SonarQube



It is an open-source tool used for continuous inspection of code quality, static analysis of code to detect bugs, code smells and security vulnerabilities on 20+ programming languages.



It provides reports on duplicated lines of code, coding standards, unit tests, coverage, code complexity, comments, bugs etc.



Benefits of SonarQube

Reduces risks in software development within a short span of time, detects bugs in the code and automatically alerts developers.

Receives all files as input , stores them in a database, analyses them with barriers and displays it in a dashboard.

It increases the productivity of Dev team by detecting duplicated & redundant code.

Dev Team receive regular feedback on quality issues and it helps them increase their programming skills.

Code quality becomes part of CI/CD & Development team.

Security flaws detection & remediation.

DevSecOps: SAST - SonarQube



1. *git clone <https://github.com/ocatak/SonarQube.git>*
2. *sudo npm install sonarqube-scanner*
3. Run SonarQube Docker image: *docker-compose up -d*
4. *GitHub: sonar-project.js*

DevSecOps: SAST - SonarQube



TSQL sample project

localhost:9000/dashboard?id=tsql.sample.project

sonarqube Projects Issues Rules Quality Profiles Quality Gates

Search for projects, sub-projects and file Log in

TSQL sample project master

Overview Issues Measures Code Activity

Quality Gate Passed

Bugs 0 A Vulnerabilities 0 A

Leak Period: since previous version started 8 hours ago

New Bugs 0 New Vulnerabilities 0

Code Smells 4h C 57 Debt 4h A New Debt 50 New Code Smells

Coverage 100% Coverage Coverage on New Code

Duplications 0.0% Duplications Duplicated Blocks Duplications on New Code

August 24, 2018, 9:49 P sonarqube-scanner

2.8.0 • Public • Published 7 months ago

XS 57 Lines of Code

No tags

Activity

Readme Explore (BETA) 9 Dependencies 72 Dependents 25 Versions

NPM module to run SonarQube/SonarCloud analyses

sonarqube-scanner makes it very easy to trigger SonarQube / SonarCloud analyses on a JavaScript code base, without needing to install any specific tool or (Java) runtime.

This module is analyzed on SonarCloud.

build passing quality gate passed maintainability A reliability A security A release v2.8.0 coverage 58%

Install

> npm i sonarqube-scanner

Weekly Downloads 198,200

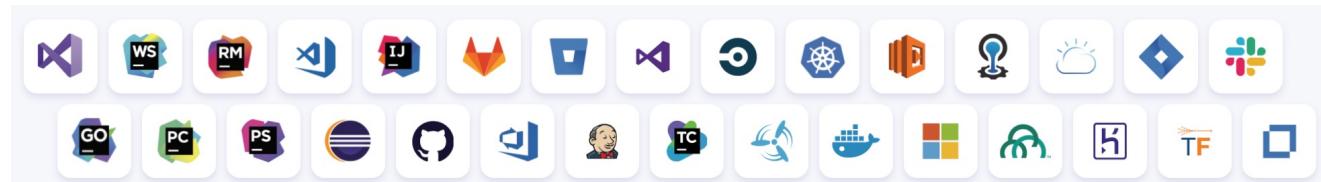
Version 2.8.0 License LGPL-3.0

Unpacked Size Total Files

In Jenkins, add a new item “Execute Shell”
npm run sonar

DevSecOps: SAST - SNYK

- Lots of Multi-Ntional Companies are protected by SNYK
 - Google
 - Microsoft
 - Salesforce
 - You can integrate it to your DevSecOps pipeline
 - Install snyk jenkins pluggin
 - Create an account in snyk
 - Create a token in snyk
 - create a synk job and input the git url for scanning,
 - add it to build step.
 - *npm install -g snyk*
 - *synk auth*



DevSecOps: SAST - SNYK

Execute shell

Command

```
cd ${WORKSPACE}
echo "snyk test && snyk monitor" > snyk.sh
chmod +x snyk.sh
/bin/bash snyk.sh || true
snyk test --json | snyk-to-html -o results.html
```

See the list of available environment variables

26 lines (23 sloc) | 341 Bytes

```
1 pipeline{
2   agent any
3   stages {
4     stage('SAST') {
5       build 'SECURITY-SAST-SNYK'
6     }
7     stage ('Build') {
8       steps {
9         echo "Build Phase"
10      }
11    }
12  }
13  stage ('Test') {
```

Jenkins

Dashboard > SECURITY-SAST-SNYK >

Back to Dashboard Status Changes Workspace Build Now Configure Delete Project HTML Report

Project SECURITY-SAST-SNYK

HTML Report Workspace Recent Changes

Permalinks

HIGH SEVERITY

HTTP Header Injection

- Package Manager: pip
- Vulnerable module: urllib3
- Introduced through: devsecops-example@0.0.0 and urllib3@1.13.1

Detailed paths

- Introduced through: devsecops-example@0.0.0 › urllib3@1.13.1

DevSecOps: SAST - SNYK

The screenshot shows the SNYK interface for a project named "ocatak/devsecops-tutorial". The project has 2 vulnerabilities found in the Dockerfile and 1 in requirements.txt. The Dockerfile has 166 High (H), 130 Medium (M), and 260 Low (L) vulnerabilities. The requirements.txt file has 162 H, 129 M, and 260 L vulnerabilities. Both were tested a few seconds ago.

File	High (H)	Medium (M)	Low (L)	Last Tested
Dockerfile	166	130	260	Tested a few seconds ago
requirements.txt	162	129	260	Tested a few seconds ago

ocatak/devsecops-tutorial:Dockerfile

Created Sun 6th Jun 2021 | Snapshot taken by snyk.io 2 minutes ago | Retest now

IMPORTED BY
F Ferhat Ozgur Catak

IMAGE TAG
3.8.1

PROJECT OWNER
⊕ Add a project owner

BASE IMAGE
python:3.8.1

SOURCE
GitHub

REPOSITORY
devsecops-tutorial

mariadb-10.3/mariadb-common - OS Command Injection

VULNERABILITY | CWE-78 | CVE-2021-27928 | CVSS 7.2 | HIGH | SNYK-DEBIAN10-MARIADB103-1087462

Introduced through

mariadb-10.3/mariadb-common@1:10.3.18-0+deb10u1, mariadb-10.3/libmariadb3@1:10.3.18-0+deb10u1 and others

Exploit maturity

Show less details ^

Detailed paths

- **Introduced through:** python@3.8.1 → mariadb-10.3/mariadb-common@1:10.3.18-0+deb10u1
Remediation: No remediation path available.
- **Introduced through:** python@3.8.1 → mariadb-10.3/libmariadb3@1:10.3.18-0+deb10u1
Remediation: No remediation path available.
- **Introduced through:** python@3.8.1 → mariadb-10.3/libmariadb-dev-compat@1:10.3.18-0+deb10u1
Remediation: No remediation path available.

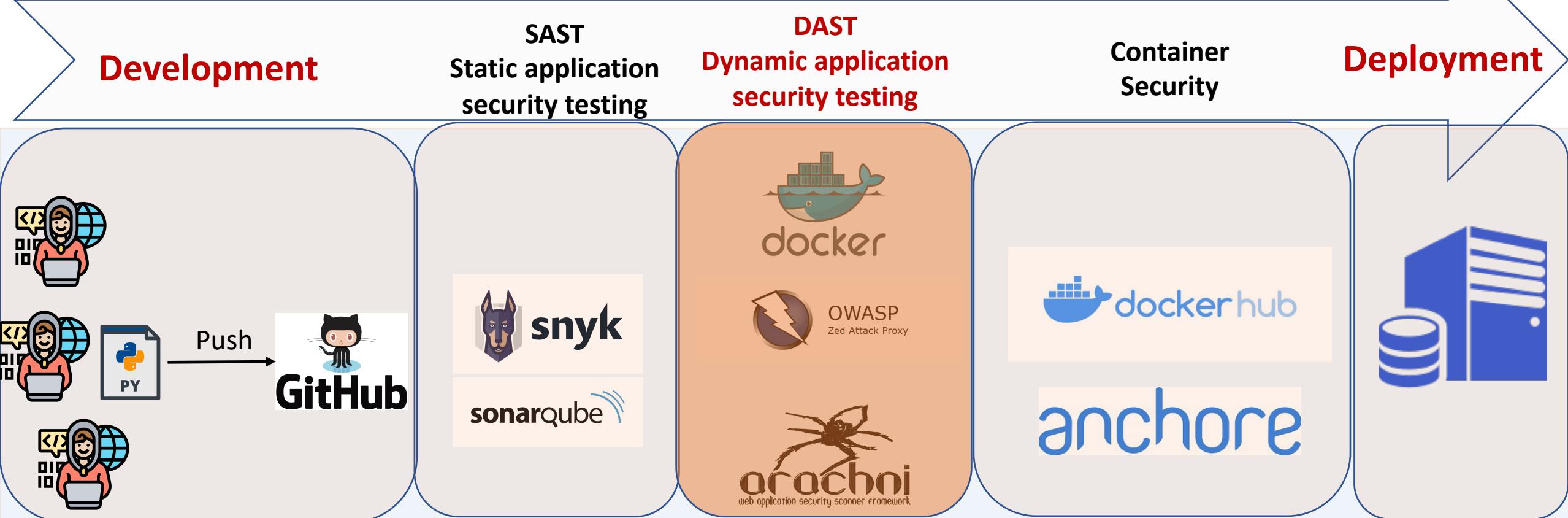
Example Web Application: DevSecOps

<https://github.com/ocatak/devsecops-tutorial>

GitHub repository for the lecture at University of Stavanger



Jenkins Pipeline



DevSecOps: DAST - ARACHNI

- Arachni is a feature-full, modular, high performance Ruby framework aimed towards helping PenTesters and administrators evaluate the security of modern web applications.
- It is free, with its source code public and available for review.
- <https://github.com/Arachni/arachni>



DevSecOps: DAST - ARACHNI

https://github.com/Arachni/arachni/releases/download/v1.5.1/arachni-1.5.1-0.5.12-linux-x86_64.tar.gz

Jenkins

Dashboard > SECURITY-DAST-Arachni > #3

Back to Project Status Changes Console Output Edit Build Information

Build #3 (Jun 6, 2021 6:22:20 PM)

No changes. Started by anonymous user

http://192.168.2.164:5000/ Generated on 2021-06-06 20:22:30 +0200

Summary

Charts Issues (4) OWASP Top 10

Issues by type, trust, and severity
(Click on the bars or line points for details on the relevant issues.)

Category	Amount of Tagged Issues
Common administration interface	1.0
Missing 'X-Frame-Options' header	1.0
Allowed HTTP methods	1.0
Interesting response	1.0

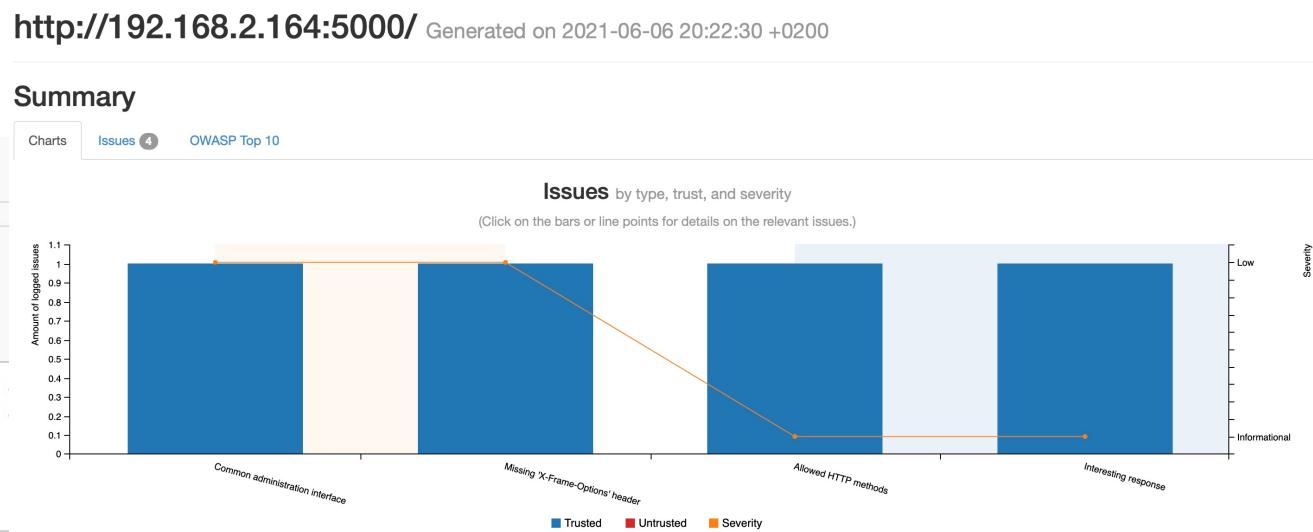
Severity: Low, Untrusted, Informational

Execute shell

Command

```
/home/devops/arachni-1.5.1-0.5.12/bin/arachni http://192.168.2.164:5000  
/home/devops/arachni-1.5.1-0.5.12/bin/arachni_reporter ${BUILD_TAG}.afr  
unzip ${BUILD_TAG}.zip  
rm -f ${BUILD_TAG}.zip
```

See the list of available environment variables



DevSecOps: DAST - ARACHNI

Summary

Charts

Issues 4

OWASP Top 10

Trusted 4

Low severity 2

Common administration interface 1

An administration interface was identified and should be reviewed.

Vector type	HTTP method	Action
 server	GET	http://192.168.2.164:5000/console

Missing 'X-Frame-Options' header 1

Clickjacking (User Interface redress attack, UI redress attack, UI redressing) is a malicious technique of tricking a Web user into clicking on something different from what the user perceives they are clicking on, thus potentially revealing confidential information or taking control of their computer while clicking on seemingly innocuous web pages.

The server didn't return an `X-Frame-Options` header which means that this website could be at risk of a clickjacking attack.

The `X-Frame-Options` HTTP response header can be used to indicate whether or not a browser should be allowed to render a page inside a frame or iframe. Sites can use this to avoid clickjacking attacks, by ensuring that their content is not embedded into other sites.

Vector type	HTTP method	Action
 server	GET	http://192.168.2.164:5000/

DevSecOps: DAST – OWASP ZAP

- Owasp-zap is one of the world's most popular free security tool and is actively maintained by hundreds of international volunteers.
- It helps to find vulnerabilities automatically with every ci/cd build.
- One of the best free tool for experienced PenTesters to be used for manual security testing.



OWASP
Zed Attack Proxy

DevSecOps: DAST – OWASP ZAP

The screenshot shows the OWASP ZAP interface with the 'Alerts' tab selected. There are two alerts listed:

- X-Frame-Options Header Not Set**
 - URL: http://192.168.2.164:5000
 - Risk: Medium
 - Confidence: Medium
 - Parameter: X-Frame-Options
 - Attack:
 - Evidence:
 - CWE ID: 16
 - WASC ID: 15
 - Source: Passive (10020 – X-Frame-Options Header)
 - Description: X-Frame-Options header is not included in the HTTP response to protect against 'ClickJacking' attacks.
- X-Content-Type-Options Header Missing**
 - GET: http://192.168.2.164:5000

Other sections visible include 'Other Info:', 'Solution:', and 'Reference:'.

Solution:
Most modern Web browsers support the X-Frame-Options HTTP header. Ensure it's set on all web pages re-framed only by pages on your server (e.g. it's part of a FRAMESET) then you'll want to use SAMEORIGIN, otherwise should use DENY. Alternatively consider implementing Content Security Policy's "frame-ancestors" directive

Reference:
<https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options>

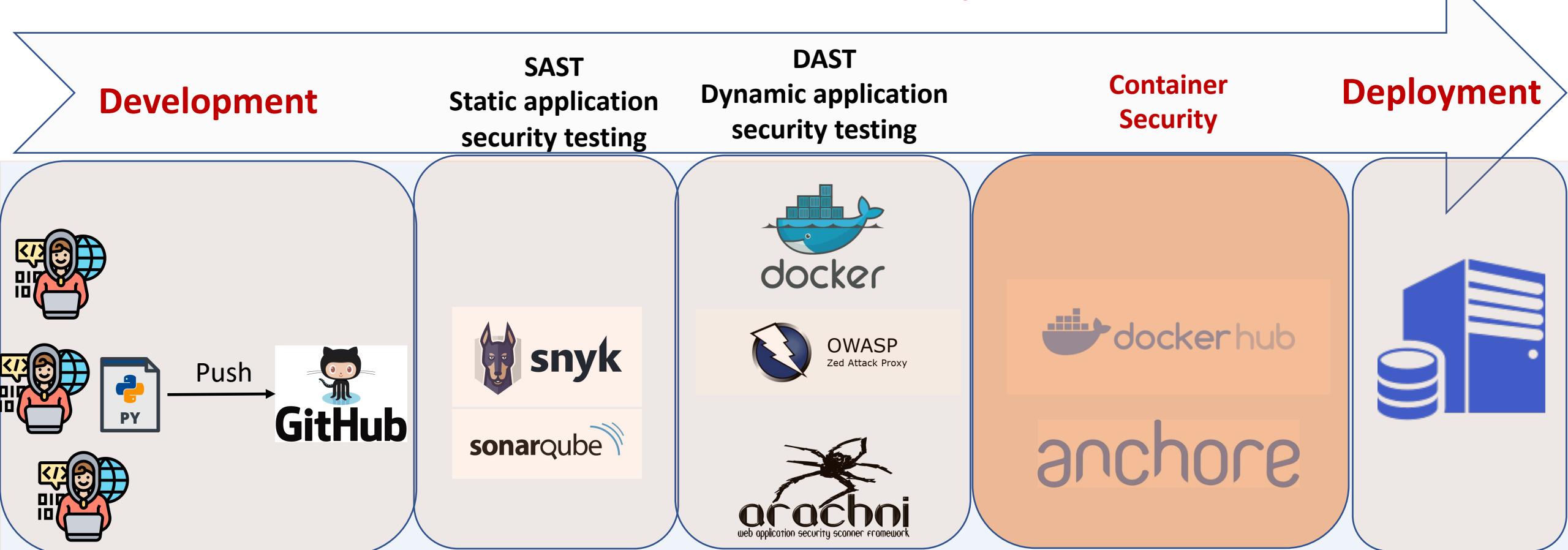
Example Web Application: DevSecOps

<https://github.com/ocatak/devsecops-tutorial>

GitHub repository for the lecture at University of Stavanger



Jenkins Pipeline



DevSecOps: Container Security - Anchore

- Anchore allows us to perform detailed analysis on container images, run queries, produce reports. Also we can define our own policies that can be used with CI/CD pipelines



DevSecOps: Container Security - Anchore

Execute shell

Command

```
echo "fcatak/devsecops_stavanger:new" ${WORKSPACE}/Dockerfile > anchore_images
cat anchore_images
```

See the list of available environment variables

Anchore Container Image Scanner

Anchore Build Options

Image list file ?

anchore_images

Fail build on policy evaluation FAIL result

Fail build on critical plugin error

Anchore Policy Evaluation Summary

Show 10 entries

Search:

Repo Tag	Stop Actions	Warn Actions	Go Actions	Final Action
docker.io/fcatak/devsecops_stavanger:new	27	77	0	STOP

Showing 1 to 1 of 1 entries

Previous

Anchore Policy Evaluation Report

Show 10 entries

Search:

Image Id	Repo Tag	Trigger Id	Gate	Trigger	Check Output	Gate Action	Whitelisted
0d0b5c8bc74cc6156bb48fb1d8c6	docker.io/fcatak/devsecops_stavanger:new	CVE-2019-25032+libunbound8	vulnerabilities	package	HIGH Vulnerability found in os package type (dpkg) - libunbound8 (CVE-2019-25032 - https://security-tracker.debian.org/tracker/CVE-2019-25032)	STOP	false
887ad8f1f34a3b44a1a64682dd819b19e32d							
0d0b5c8bc74cc6156bb48fb1d8c6	docker.io/fcatak/devsecops_stavanger:new	CVE-2019-25033+libunbound8	vulnerabilities	package	HIGH Vulnerability found in os package type (dpkg) - libunbound8 (CVE-2019-25033 - https://security-tracker.debian.org/tracker/CVE-2019-25033)	STOP	false
887ad8f1f34a3b44a1a64682dd819							

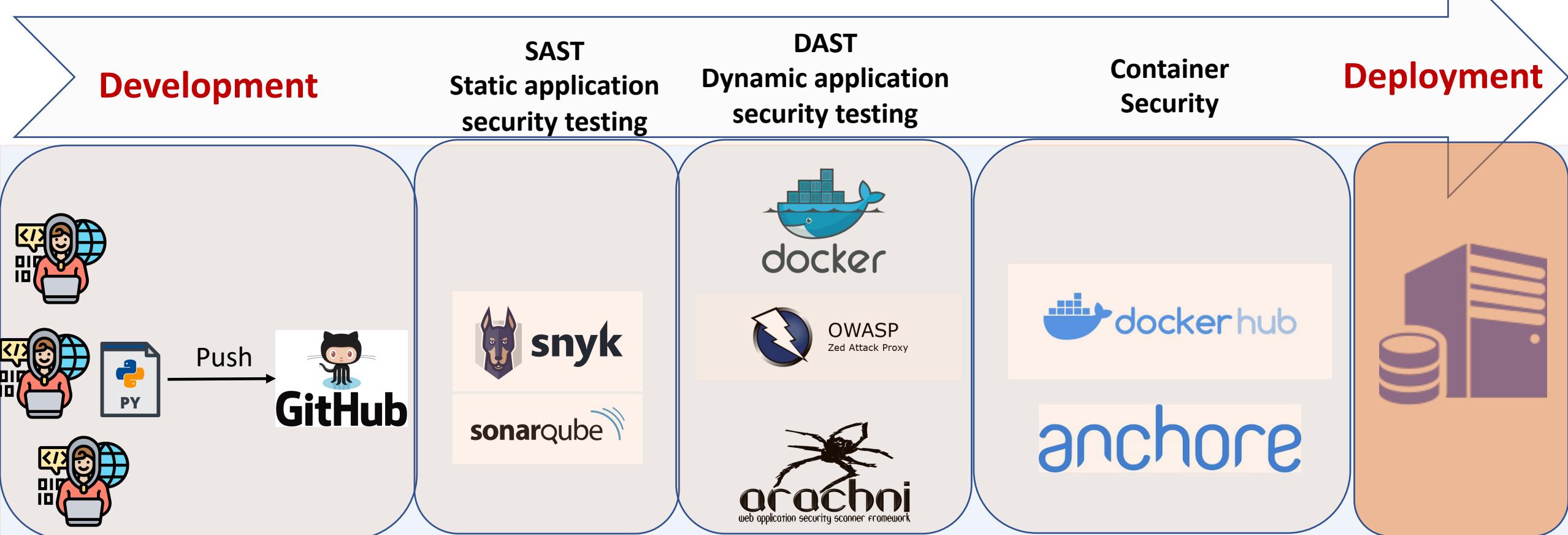
Example Web Application: DevSecOps

<https://github.com/ocatak/devsecops-tutorial>

GitHub repository for the lecture at University of Stavanger



Jenkins Pipeline



What We Learned?

- Paradigm-shifting from time consuming Penetration testing Activities to DevSecOps
- PenTesters use same/similar tools (Kali, Sqlmap, Arachni, Nmap, Metasploit etc) to perform a security testing to find the vulnerabilities
- We can integrate them into our CI/CD Pipelines
- Speed-Up security checks

Future Challenges

- Complex Tool Integrations
 - come from various vendors.
 - Adding security tools makes things even more complex.
 - time-consuming, labor-intensive approach is incompatible with DevOps' high-speed,
- Shifting security left in the SDLC
 - Security principles apply to overall organization and not just to security engineering teams.
 - End-of-cycle security testing results in delays and vulnerabilities left in production



Search or jump to...

/ Pull requests Issues Marketplace Explore

ocatak / devsecops-tutorial

Unwatch ▾ 1

Unsta

Code Issues Pull requests Actions Projects Wiki Security 2 Insights Settings

⚠ We found potential security vulnerabilities in your dependencies.

Only the owner of this repository can see this message.

See Dependabot alerts

master ▾

2 branches

0 tags

Go to file

Add file ▾

Code ▾

ocatak SonarQube

anchore-security

anchore-security

docker-jenkins

Jenkins

templates

README.md

Dockerfile

docker

Jenkinsfile

SNYK

README.md

README

arachni.sh

docker

docker-compose.yml

docker

512af74 5 days ago 23 commits

Thank You

F. Ozgur Catak

f.ozgur.catak@gmail.com

<https://www.ozgurcatak.org>

6 days ago

7 days ago

7 days ago

About

DevSecOps best practices
vulnerable Flask based web
application

python docker jenkins
devops course arachni
devsecops

Readme

Releases

No releases published
[Create a new release](#)

Packages

No packages published
[Publish your first package](#)